VU VRIJE UNIVERSITEIT AMSTERDAM

VU Research Portal

Spell sequences, state proximities and distance metrics

Elzinga, C.H.; Studer, M

published in
Sociological Methods and Research
2015

DOI (link to publisher)
10.1177/0049124114540707

document version
Publisher's PDF, also known as Version of record

Link to publication in VU Research Portal

citation for published version (APA)
Elzinga, C. H., & Studer, M. (2015). Spell sequences, state proximities and distance metrics. *Sociological Methods and Research*, *44*(1), 3-47. https://doi.org/10.1177/0049124114540707

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

E-mail address:
vuresearchportal.ub@vu.nl

Download date: 04. May. 2023

Article

Spell Sequences, State Proximities, and Distance Metrics

Sociological Methods & Research
2015, Vol. 44(1) 3-47
© The Author(s) 2014
Reprints and permission:
sagepub.com/journalsPermissions.nav
DOI: 10.1177/0049124114540707
smr.sagepub.com

Cees H. Elzinga[1] and Matthias Studer[2]

Abstract

Because optimal matching (OM) distance is not very sensitive to differences in the order of states, we introduce a subsequence-based distance measure that can be adapted to subsequence length, to subsequence duration, and to soft-matching of states. Using a simulation technique developed by Studer, we investigate the sensitivity, relative to OM, of several variants of this metric to variations in order, timing, and duration of states. The results show that the behavior of the metric is as intended. Furthermore, we use family formation data from the Swiss Household Panel to compare a few variants of the new metric to OM. The new metrics have been implemented in the freely available TraMineR-package.

Keywords

sequence analysis, OM, subsequence, soft-matching, duration-weighting

Introduction

Sequence analysis is the generic name for a variety of methods that subserve the analysis of state sequences like life courses and job careers.[1] Today,

[1] VU University, Amsterdam, the Netherlands
[2] University of Geneva, Geneva, Switzerland

Corresponding Author:
Cees H. Elzinga, VU University, De Boelelaan 1081, Amsterdam, 1081 HV, the Netherlands.
Email: c.h.elzinga@vu.nl

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

sequence analysis has become one of the standard toolboxes for those who analyze sequence data, and sophisticated, user-friendly software for such methods is freely available (e.g., Brzinsky-Fay, Kohler, and Luniak 2006; Elzinga 2009; Gabadinho et al. 2011).

To compare sequences, one needs a measure of distance or similarity between pairs of sequences and by far the most frequently used metric to generate such distances is the so-called optimal matching (OM) metric.[2] The OM metric expresses distances in terms of the minimum cost of a sequence of edit operations that turns one sequence into an exact copy of the other sequence. In the sequel, we will write d_{OM} to denote this metric.

Ample descriptions of the metric and the associated algorithm can be found in numerous sources, for example, in Clote and Backofen (2000), in Martin and Wiggins (2009) and in Sankoff and Kruskal (1983). Largely motivated by the problems of determining the weight or cost of the edit operations involved, many variants of the metric have been proposed, some quite general (e.g., Gauthier et al. 2009; Halpin 2010; Moen 2000), others more application-specific like in, for example, Lesnard (2008). For a comprehensive overview of OM variants, the reader is referred to Studer (2012).

For various reasons, the use of OM in the social sciences has been widely criticized, most notably in Settersten and Mayer (1997), Dijkstra and Taris (1995), Wu (2000), Levine (2000), Elzinga (2003), and Lesnard (2008). The first major point of critique has been that, in the social sciences, edit operations have no interpretation; they cannot be interpreted as spontaneous or selection-driven mutations like in microbiology. However, OM can be interpreted in a way that does not involve edit operations at all but instead refers to the concept of a longest common subsequence (lcs). When we suppose that the cost of inserting or deleting any character equals 1 and the cost of substituting one character for another, distinct character equals 2 (one deletion followed by one insertion), we have that

$$d_{\mathrm{OM}} = \ell(x) + \ell(y) - 2\ell(\mathrm{lcs}(x,y)), \tag{1}$$

wherein $\ell(x)$ denotes the length of sequence x. Hence, the OM distance equals the number of characters in both sequences that do *not* belong to an lcs of the pertaining sequences x and y. If we interpret $\mathrm{lcs}(x,y)$ as a "common backbone" or "common narrative," then d_{OM} decreases with the relative length of that common backbone. So, we don't really need to interpret the edit operations that are often used to define the OM distance and that explain the logic of the OM algorithm. In case the edit costs have been set differently, the interpretation of OM distance is in terms of an lcs that does not equally weight all states.

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

The second major criticism of OM pertains to the fact that often there is no objective way of establishing the edit costs of the edit operations. Various, more or less sophisticated, ways of deriving edit cost from state-transition frequencies have been devised (see, e.g., Gauthier et al. 2009), but these methods do not resolve the basic issue: establishing the proximity or similarity of states and how this similarity can be derived and operationalized from social science theory. This matter has not been resolved and cannot be resolved within the framework of sequence analysis (Studer 2012). On the other hand, Hollister (2009) and Studer (2012) propose promising strategies to establish state proximities through scaling strategies that are independent of sequence analysis. Despite the trouble we have in finding acceptable ways of establishing state proximities or state similarities, we cannot do without them. To illustrate this point, we consider three toy sequences from the domain of family formation, using the states Single, Unmarried cohabitation, Married, and Married with Children:

x:	S	M	MC
y:	S	U	MC
z:	S	S	MC

Whatever metric we use, we should find that *x* is closer to *y* than it is to *z*, simply because the state U is more similar to the state M than to the state S. The second reason to consider state similarities is that it is a key feature to compute multichannel distances (Pollock 2007; Gauthier et al. 2010).

According to some authors, it is more convenient to invest in defining OM costs rather than moving to a more analytical definition of the dissimilarities. Perhaps that would be a viable strategy when the issue of establishing edit cost would be the only challenge for OM in particular or sequence analysis in general.

However, not so well known or ignored is the fact (Elzinga 2003; Studer 2012) that OM is not very sensitive to differences in the order of the states of a pair of sequences. As an example, we consider the three toy sequences below representing careers, using state **d** for "director," **m** for "manager," and **e** for "employee":

x:	**e**	**m**	**m**	**m**	**m**	**m**	**m**	**m**	**d**
y:	**d**	**m**	**m**	**m**	**m**	**m**	**m**	**m**	**e**
z:	**e**	**e**	**e**	**e**	**m**	**d**	**d**	**d**	**d**

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

According to the OM metric, sequences x and y are the closest pair because they share a long "common narrative": the seven time units spent in the **m** state. However, sequences x and y reflect opposite career dynamics: x can be interpreted as an "ascending" career, y is a "descending" one. There are two reasons for this lack of sensitivity to ordering. First, OM is context-insensitive: Each state is handled separately without considering previous or subsequent states (Halpin 2010; Hollister 2009). Edit operations are applied on symbols in strings, irrespective of their context (previous and subsequent states notably). Hence, the dynamics of the trajectory are not explicitly handled by OM. Second, the only way to handle duration through OM is by repeating the same state with a frequency that corresponds to the number of time units intended. In the social sciences, trajectories are often coded with a few relatively long spells. As a result, the "common" backbone are often strongly linked with the total time spent in some states, ignoring the underlying dynamics encoded with small spells.

This lack of sensitivity to ordering is problematic since sequence analysis is about differences between categorical time series where event orderings or state orderings are defining the sequences. Moreover, the ordering of the states reflects the internal dynamics of a trajectory, one of the important aspects that sequences analysis claims to take into account. Therefore, Elzinga (2003, 2005) proposed a distance metric that is based on a subsequence-based vector space. Subsequences allow analyzing states in their contexts and thus the dynamic of the trajectory. In our example, the subsequence *ed* (employee–director) provides essential information to study the dynamics of the trajectory. Simulations presented by Studer (2012) demonstrate that subsequence-based metrics are much more sensitive to differences in state orderings.

However, Elzinga's metric does not allow for different state proximities: All states are considered as equally different (see also Hollister 2009). Therefore, we propose a very flexible generalization of Elzinga's subsequence-based metric that does handle such state proximities. The metric has a number of interesting properties that are best explained through representing sequences as vectors in a vector space with a Euclidean norm.

As said before, one limitation of OM is the way it handles time. This limitation is caused by the fact that OM counts edits applied to symbols in a string and has no inherent mechanism to deal with quantities like duration. Therefore, the observation that someone was unemployed (U) for 10 months and then found herself a job for the next 30 months has to be translated into a sequence of the form

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

$$\underbrace{UU\ldots U}_{10}\underbrace{EE\ldots E\ldots E}_{30},$$

that is, as a sequence of 40 observations of states. This mapping of durations to strings of states severely limits the way in which time or duration can be handled (see, e.g., Halpin 2010). However, representing the observations as

$$(U, 10)(E, 30),$$

that is, as two states, one with a duration of 10 months and one with a duration of 30 months may seem more natural. Choosing a different time scale, say years, would then invite to write $(U, .833)(E, 2.50)$ or $(U, 300)(E, 900)$ when the scale were days instead of years or months.

Formally, a state sequence like for example, $x = abac$ is a concatenation of states from some alphabet $\Sigma = \{\lambda, a, b, \ldots\}$ and a k-spell sequence consists of a pair

$$(x, \mathbf{t}) = (x_1, t_1)(x_2, t_2)\ldots(x_k, t_k),$$

that is, a state sequence and a numerical sequence. Using a metric that can handle time as a quantity that can be separated from the states would allow for a more sophisticated treatment of the time dimension (see, e.g., Abbott and Hrycak 1990; Halpin 2010).

The purpose of this article is to discuss a family of distance measures that is quite sensitive to differences in the sequencing of the states or events, that allows for proper duration handling and time transforms, and that uses state proximities. We will discuss the metrics and we will demonstrate their sensitivity, relative to OM. Finally, we will demonstrate their practical use in an application to family formation.

Thereto, the next section introduces the representation of sequences through feature vectors, the features being subsequences. The third section then discusses soft-matching, the use of state proximities, and the required transform of the vector space. The fourth section discusses spell sequences: sequences where duration is treated as a property of the states. In the fifth section, we discuss the unifying framework of a feature vector representation and in the sixth section, we assess the sensitivity of the metrics to differences in sequencing, timing, and duration of the pertaining states. Finally, in the seventh section, we apply the newly introduced metrics to family formation data and compare the results with those obtained using OM. In the eight section, we discuss our findings and the merits thereof.

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

Sequences as Vectors

Vectors and Distances

What makes vector representations so interesting? Vector representations are interesting because once we have vectors, there is a whole family of distance measures that are proper metrics in the sense that these distance measures satisfy the axioms of a metric:

D1: $d(x,x) = 0$,
D2: $d(x,y) > 0$,
D3: $d(x,y) = d(y,x)$,
D4: $d(x,z) \leq d(x,y) + d(y,z)$.

D1 states that an object has one location only and D2 states that two distinct objects cannot be in the same location. D3, the symmetry axiom, states that direction does not affect distance, and D4, the so-called triangle inequality, states that "a detour takes at least as much time" or, put differently, that when two objects (x and z) are close to a third object (y), they cannot be remote from each other.

The triangle inequality is not only important because it formulates an intuitive property of a quantified space. The triangle inequality also ensures that objects can be located with respect to each other without other objects being involved. If the triangle inequality would not hold, the observation of $d(x,z)$ would not be very meaningful if there could exist some (unobserved) y such that $d(x,y) + d(y,z) < d(x,z)$. When working with groups of sequences (such as clusters), the triangle inequality ensures that a particular observation y does not artificially create an homogeneity in the group by "attracting" objects that would be considered as very distant when y would not be observed.

Finally, the triangle inequality ensures that the space exhibits a certain regularity or smoothness in the sense that at least some of its properties are invariant in all directions. If this were not true, the space, the representation of the sequences in a distance matrix, would not be very meaningful. Let us illustrate this remark: Imagine that we have observed a set of sequences $\{x,y,z,\ldots\}$, say $N = 1,000$ sequences, and that we have somehow established distances between the pairs of sequences. Now suppose that we add a new observation, a new sequence p, to our data set. We may not know yet how to localize this p in the spatial representation of the N sequences. However, if the space is metric in the sense that the distances satisfy the axioms D1 to D4, we know that we have

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

$$d(p,x) \leq d(p,y) + d(y,x), \tag{2}$$

for *all* pairs (x,y) of known sequences. So, given a metric space with N objects, the distance of the new object to all known objects must satisfy $\binom{N}{2} = N(N-1)/2$ restrictions of the form of equation (2). Even if equation (2) is satisfied for all pairs (x,y), this does not imply that

$$d(x,y) \leq d(x,p) + d(p,y), \tag{3}$$

holds for all pairs (x,y) too. Thus, the number of restrictions, that is, inequalities, that the new sequence must satisfy, equals $2\binom{N}{2} = N(N-1)$. For a moderate data set of say, thousand sequences, this amounts to almost one million inequalities to be satisfied.

So the triangular inequality severely limits the location of the new sequence p in sequence space: The distances to p cannot be wildly at variance with what we expect on the basis of what we already know about the space. Furthermore, when we add p to the space, that is, when we enlarge our knowledge from N to $N+1$ sequences, the next sequence q to be added will have to satisfy $2\binom{N+1}{2} = 2\binom{N}{2} + 2N$ inequalities. So, the more sequence distances we know, the more we know, in terms of the number of inequalities to satisfy, about the structure of the space: Adding more sequences and gauging their distances makes sense. Conversely, if we drop the triangular inequality as a requirement, gathering new data would not add much to our understanding of sequences through a spatial representation. Therefore, it is essential that the procedures with which we assign (distance-)numbers to pairs of sequences ascertain that the triangular inequality is satisfied. Moreover, many distance analysis methods such as clustering algorithms or discrepancy analysis are based on these axioms.

We know (see, e.g., Clote and Backofen 2000) that the OM distance d_{OM} satisfies all four axioms, provided that the edit cost function is a metric over the state alphabet. So, the OM algorithm as such does not guarantee a proper metric; one needs a metric cost function as well. Examples of metric and non-metric cost functions are shown in Table 1. However, many of the variants of OM that somehow (dynamically) adapt the standard edit cost matrix may lead to violations of the triangle inequality (Studer 2012).

Once vectors are available, it is easy to calculate Euclidean distance d_E: for vectors $\mathbf{x} = (x_1, x_2, \ldots)$ and $\mathbf{y} = (y_1, y_2, \ldots)$, we have that

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

Table 1. Illustration of the Metric Properties of the OM Standard Edit Cost Matrix.

	λ	*w*	*x*	*y*	*z*		λ	*w*	*x*	*y*	z
λ	0	1	1	1	1	λ	0	1	1	1	1
w	1	0	2	2	2	*w*	1	0	2	2	2
x	1	2	0	2	2	*x*	1	2	0	1.5	4
Y	1	2	2	0	2	*y*	1	2	1.5	0	2
z	1	2	2	2	1	*z*	1	2	4	2	0

Note. OM = optimal matching. The reader verifies that, in the left hand matrix, $c(\cdot,\cdot)$ satisfies the axioms D1–D4. For example, we have that $c(w,y) \leq c(w,x) + c(x,y)$ for every x. However, perturbations of this matrix may easily lead to violations of the triangle inequality D4. This is shown in the right hand matrix, where we have $c(x,z) > c(x,y) + c(y,z)$.

$$d_E(\mathbf{x},\mathbf{y}) = \left(\sum_i |x_i - y_i|^2\right)^{\frac{1}{2}}, \tag{4}$$

$$= \sqrt{\mathbf{x}'\mathbf{x} + \mathbf{y}'\mathbf{y} - 2\mathbf{x}'\mathbf{y}}. \tag{5}$$

In this article, we will only be concerned with Euclidean distances since they can be evaluated without even "knowing" or constructing the vectors explicitly: From equation (5), we see that we can evaluate the distances, provided that we have access to the values of the inner products $\mathbf{x}'\mathbf{x}$, $\mathbf{y}'\mathbf{y}$, and $\mathbf{x}'\mathbf{y}$. The vectors that we will construct will appear to have extremely high dimension but fortunately, there exist efficient algorithms, so-called kernels (see, e.g., Schölkopf and Smola 2002), that evaluate inner products without requiring the coordinate values of the pertaining vectors.

Finally, vector spaces are very attractive to work with because they have been amply studied in linear algebra (see, e.g., Meyer 2000) and much of this knowledge is exploited in the standard multivariate statistical models.

In the next two subsections, we will discuss how to construct vectors from sequences. Essentially, this is a new presentation of Elzinga's proposals as discussed in Elzinga (2003, 2005). These new presentations allow us to discuss vector representations without referring to algorithms for the evaluation of vector products. In the sections Mapping Embedding Frequency, we discuss an easy extension and in the sections State Matching and Inner Product Spaces and Spell Sequences: Handling Durations, we exploit the representation to discuss more advanced issues like the handling of time and state matching.

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

The Basic Representation

We will construct vectors from sequences through using the concept of "subsequence," so we begin with elaborating on this concept. For a more formal treatment, the reader is referred to for example, Apostolico and Cunial (2009); Crochemore, Hancart, and Lecroq (2007); or Elzinga, Rahmann, and Wang (2008).

Consider the toy sequence $x = x_1x_2x_3x_4 = abac$ over a three-state alphabet $\Sigma = \{a, b, c\}$. In this and the next subsections, we will use this toy sequence to illustrate the basic principles of constructing vectors from sequences and some elementary variants thereof.

We may take any nonnegative number of states from x and we will then be left with a subsequence of x: a subsequence u of states that have the same order in x and we will write $u \sqsubseteq x$ to denote such fact. For example, when we take out the a's from x, we will be left with $u = bc$, one of the four 2-long subsequences of x. At most, we can take away all states from x and we will then be left with an empty sequence for which we use the symbol λ. We might also take the smallest nonnegative number of states from x, zero states, and we would be left with x itself and hence we conclude that $x \sqsubseteq x$. The reader might want to verify that x has 13 distinct subsequences, including λ and x itself.

We will now use the concept of subsequence to construct a vector representation $\mathbf{x}$ for the sequence x. We do this by first defining coordinates that correspond to all possible sequences that can be constructed from the alphabet Σ and then construct binary vectors by setting those coordinates to 1 that correspond to sequences that occur as a subsequence in $x = abac$

u:	a	b	c	aa	ab	...	cc	aaa	...	aba	...
$r(u)$:	1	2	3	4	5	...	12	13	...	16	....
$\mathrm{x}_{r(u)}$:	1	1	1	1	1	...	0	0	...	1	...

Formally, from Σ, we construct the set Σ^* of all sequences that are constructible from Σ and we fix the order of the elements of Σ^*, say in lexicographical order. Then we map the ordered sequences to the nonnegative integers Z^+, that is, each sequence $u \in \Sigma^*$ is mapped to a unique[3] integer $r(u) \in Z^+$ and we use these integers to index the coordinates of the vectors. So, for each sequence x, we construct a binary vector $\mathbf{x} = (\mathrm{x}_1, \mathrm{x}_2, \ldots)$ such that

$$\mathrm{x}_{r(u)} = \begin{cases} 1 & \text{if } u \sqsubseteq x \\ 0 & \text{otherwise} \end{cases}. \tag{6}$$

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

This construction characterizes strings by their subsequences and the resulting vectors are also called "feature vectors," the subsequences being treated as features of the sequence.

The inner product $\mathbf{x}'\mathbf{y} = \sum_i \mathrm{x}_i\mathrm{y}_i$ counts the number of distinct common subsequences and therefore, the squared Euclidean distance $d^2(\mathbf{x}, \mathbf{y}) = \mathbf{x}'\mathbf{x} + \mathbf{y}'\mathbf{y} - 2\mathbf{x}'\mathbf{y}$ is measured in terms of the number of distinct subsequences that are unique to either x or y. Intuitively, sequences are more similar when they have more features, more subsequences, in common.

In practice, this is a very appealing feature. It means that, using a cluster analysis, sequences grouped together will share the same subsequences. In a discrepancy analysis (Studer et al. 2011), a test would be significant if the subsequences of one group are significantly different from those of the other one. This would be similar to using multivariate analysis of variance (MANOVA) in the subsequence space.

The vectors so constructed have a countably infinite dimension since the index function r is a bijection from Σ^* to the nonnegative integers. Therefore, evaluating the inner product $\mathbf{x}'\mathbf{y}$ through $\mathbf{x}'\mathbf{y} = \sum_i \mathrm{x}_i\mathrm{y}_i$ is not feasible and one needs a kernel function (e.g., Elzinga et al. 2008; Schölkopf and Smola 2002) to evaluate $\mathbf{x}'\mathbf{y}$ without explicitly constructing the vectors.

The representation in equation (6) is very simple in the sense that it just uses the presence or absence of subsequences to represent the sequences and thus it is tempting to use substantially more interesting properties of the subsequences (provided that kernel functions exist that evaluate inner products of the resulting vectors). This is exactly what we will do in the following subsections: Define more sophisticated properties of the subsequences and use these to modify the distance measure according to its application.

Mapping Embedding Frequency

Returning to our toy sequence $x = abac$, we observe that the subsequence $u = ac$ is embedded twice in x: as x_1x_4 and as x_3x_4. We denote this fact by writing $|x|_u = 2$.

Unfortunately, the sequence "Imprisoned, Probation, Convicted" is a subsequence that is embedded more than once in many a criminal career and we know that frequency of embedding of such subsequences is a relevant feature when comparing criminal careers. Similarly, the embedding frequency of the subsequence "Unemployed, Vocational Training, Employed" is an interesting feature of labor market careers.

From the previous examples, we conclude that taking embedding frequency into account when comparing sequences may be a sensible thing to

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

do and it is easily accomplished by constructing vectors through defining coordinates according to

$$x_{r(u)} = \begin{cases} |x|_u & \text{if } u \sqsubseteq x \\ 0 & \text{otherwise} \end{cases}. \tag{7}$$

To interpret the meaning of $\mathbf{x}'\mathbf{y}$, it is convenient to introduce a new concept: the set $\mathcal{S}(x,y)$ of distinct common subsequences of the sequences x and y. When using the representation (7), evaluating the inner product $\mathbf{x}'\mathbf{y}$ amounts to calculating

$$\mathbf{x}'\mathbf{y} = \sum_i x_i y_i \tag{8}$$

$$= \sum_{u \in \mathcal{S}(x,y)} |x|_u \cdot |y|_u, \tag{9}$$

for if $u \notin \mathcal{S}(x,y)$, either $|x|_u = 0$ or $|y|_u = 0$ or both. So, we interpret the value of $\mathbf{x}'\mathbf{y}$ as "the number of matching subsequences (NMS)" of x and y: For each $u \sqsubseteq x$, there exist $|y|_u$ matches in y hence the total number of matches equals $|x|_u \cdot |y|_u$ and we add these quantities for all $u \in \mathcal{S}(x,y)$ when calculating $\mathbf{x}'\mathbf{y}$. The similarity and distance as proposed in Elzinga (2003) and Elzinga (2005) are in fact derived from the representation (7).

Mapping Subsequence Lengths

Most people share, in most kinds of careers, a lot of single states. For example, when studying family formation careers, we know that most people started living with their parents, then become parents themselves and before that, live together with a partner. Similarly, most people go to school before starting to work, and so on. So, we may expect that many careers share the same *short* subsequences. Therefore, when counting the number of common or matching subsequences, that is, when using representations (6) or (7), it might be interesting to weight the counts according to the length of the subsequences by some convex function $L(\ell(u))$ of the subsequence lengths $\ell(x)$. This can be accomplished by the representation

$$x_{r(u)} = \begin{cases} \sqrt{L(\ell(u))} & \text{if } u \sqsubseteq x \\ 0 & \text{otherwise} \end{cases}. \tag{10}$$

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

The square root is arising since we are interested in evaluating an inner product $\mathbf{x}'\mathbf{y}$ resulting in the proper weighting of the counts. When weighting the NMS, we now attain

$$\mathbf{x}'\mathbf{y} = \sum_{u \in \mathcal{S}(x,y)} \sqrt{L(\ell(u))}\sqrt{L(\ell(u))} \tag{11}$$

$$= \sum_{u \in \mathcal{S}(x,y)} L(\ell(u)). \tag{12}$$

For example, by setting $L(a) = (a-1)^p$ for $a \geq 1$ and $p > 1$, one would ignore common single states and assign progressively more weight to longer subsequences.

State Matching and Inner Product Spaces

In this section, we will extend the methods dealt with to using nonperfect matchings between states and, consequently, matchings between subsequences. All of the methods discussed so far construct vectors from sequences in order that the inner product of such vectors is equivalent to a weighted count of the common subsequences. Such counts are then used to construct distances and similarities. Incorporating subsequence matchings will allow us to also count nonperfect matches and weight these appropriately.

We first have to define such matchings and this is the subject of the first subsection. Once defined, we will have to investigate how we can use them. This is nontrivial since the standard inner product counts the common subsequences that are perfect matches: The vector coordinates are indexed by the set of distinct subsequences and hence the inner product $\mathbf{x}'\mathbf{y} = \sum_i \mathrm{x}_i\mathrm{y}_i$ counts the number of common subsequences, "common" meaning that an exact copy of a particular subsequence occurs in the other sequence too. Therefore, we will need to extend the notion of an inner product in order to allow for counting not only exact copies but also approximate matches. This counting problem will be dealt with in the second subsection.

Matchings

We already argued that generating meaningful distances between sequences is not well possible without assessing the similarity or substitutability of the states or events involved. On the other hand, the actual assessment of such quantities is highly dependent upon the subject matter of the sequences so,

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

in a methodological essay, it is not possible to detail the evaluation of state similarity. On the other hand, we have seen authors (e.g., Chen, Ma, and Zhang 2009; Elzinga 2014; Elzinga et al. 2011; Emms and Franco-Penya 2012; Gower 1971; Gower and Legendre 1986; Tversky 1977; Wang 2006) dealing the general issue of similarity measures and their properties. However, a detailed account of their ideas is far beyond the scope of this article. Here, it suffices to state that we assume that we have somehow defined or constructed similarities between the states of the alphabet. With an alphabet $\Sigma = \{\sigma_1, \ldots, \sigma_d, \}$, this implies that we have a $(d \times d)$-matrix $\mathbf{M} = \left(m_{ij}\right)$ such that m_{ij} denotes the degree of matching between states σ_i and σ_j. We assume that the matchings satisfy $0 \leq m_{ij} = m_{ji} < 1$ and $m_{ii} = 1$. Hence, $\mathbf{M}$ is a positive symmetric matrix of the form

$$\mathbf{M} = \begin{pmatrix} 1 & \ldots & m_{1d} \\ \vdots & \ddots & \vdots \\ m_{d1} & \ldots & 1 \end{pmatrix}.$$

For example, for the alphabet of living arrangements $\Sigma = \{S, U, M, UC, MC\}$, we might have that $\mathbf{M}$ looks like (omitting zeros)

$$\begin{array}{c} S \\ U \\ M \\ UC \\ MC \end{array} \begin{pmatrix} 1 & & & & \\ & 1 & .8 & & \\ & .8 & 1 & & \\ & & & 1 & .9 \\ & & & .9 & 1 \end{pmatrix},$$

implying that being Married is very similar to living in Unmarried cohabitation and that this similarity even increases when there are children in the household. We not only compare states, we also compare sequences and we express the degree of matching $m(x, y)$ of two equally long sequences x and y as the product of the matching coefficients of the consecutive states:

$$m(x, y) = \prod_i m(x_i, y_i). \tag{13}$$

For sequences x and y of unequal length, we set $m(x, y) = 0$. The reader notes that for two identical sequences, we always obtain $m(x, y) = 1$ and that, when two sequences have two states x_i and y_i with $m(x_i, y_i) = 0$, we obtain $m(x, y) = 0$. For the sequences of living arrangements, we obtain

$$m(x = S\ U\ UC, y = S\ M\ MC) = 1 \cdot 0.8 \cdot 0.9 = 0.72.$$

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

$$\mathbf{M} = \begin{pmatrix} 1 & p & q \\ p & 1 & r \\ q & r & 1 \end{pmatrix}, \mathbf{M}^* = \begin{pmatrix} \mathbf{M} & \dots & \mathbf{0} & \dots & \mathbf{0} & \dots \\ \vdots & \mathbf{M} & p\mathbf{M} & q\mathbf{M} & \vdots & \\ \mathbf{0} & p\mathbf{M} & \mathbf{M} & r\mathbf{M} & \mathbf{0} & \dots \\ \vdots & q\mathbf{M} & r\mathbf{M} & \mathbf{M} & \vdots & \\ & & \vdots & & \mathbf{M} & \dots \\ \mathbf{0} & \dots & \mathbf{0} & \dots & \vdots & \ddots \\ \vdots & & \vdots & & q^2\mathbf{M} & \end{pmatrix}$$

Figure 1. Structure of the matrix $\mathbf{M}^*$, given an alphabet $\Sigma = \{a, b, c\}$ and lexicographic ordering of Σ^*. $m(a, b) = p$, $m(a, c) = q$ and $m(b, c) = r$.

Most importantly, we observe that given the matrix $\mathbf{M}$ of state matchings, the matching of any pair of sequences, of whatever lengths, can be determined through using equation (13), resulting in a matrix $\mathbf{M}^*$. This matrix has entries that can be indexed by all sequences that can be constructed using the pertaining alphabet and has a structure that is illustrated in Figure 1 for an alphabet of just three states a, b, and c where the sequences are ordered lexicographically to index the entries.

The reader observes that, due to the multiplicative structure of the matchings, the matrix has a very regular structure. The reader also notes that the submatrix containing the single-state matchings regularly reoccurs, in Figure 1 as $\mathbf{M}$ in the upper left corner of $\mathbf{M}^*$. It is not very difficult but beyond the scope of this article to prove that $\mathbf{M}^*$ is singular only if this upper-left submatrix is singular. This is equivalent to saying that the inverse of $\mathbf{M}^*$ exists whenever the determinant of this submatrix is positive. In the example of Figure 1, this implies that $(\mathbf{M}^*)^{-1}$ exists if $|\mathbf{M}| > 0$.

Generalizing the Standard Inner Product

So far, we have discussed a basic vector representation of sequences that utilizes more or less sophisticated properties of the subsequences. The distance $d(\mathbf{x}, \mathbf{y})$ corresponds to the length or "norm" of the line $\| \mathbf{x} - \mathbf{y} \|$ between $\mathbf{x}$ and $\mathbf{y}$ and we evaluated lengths of lines between vectors as

$$d(x,y) = \| \mathbf{x} - \mathbf{y} \| = \sqrt{\sum_i (x_i - y_i)^2} = \sqrt{\mathbf{x}'\mathbf{x} + \mathbf{y}'\mathbf{y} - 2\mathbf{x}'\mathbf{y}}, \qquad (14)$$

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

wherein $\mathbf{x}'\mathbf{y}$ denotes the inner product $\sum_i \mathrm{x}_i\mathrm{y}_i$. However, this way of defining the norm and inner product are not very helpful when we want to employ state similarities in evaluating distances. The reason is that the coordinates of the vectors refer to separate (concatenations of) states and evaluating $\mathbf{x}'\mathbf{y} = \sum_i \mathrm{x}_i\mathrm{y}_i$ is confined to comparing the values on the *same* coordinates as indexed by i. Therefore, we now turn our attention to a more general way of defining inner products and vector norms (see, e.g., Meyer 2000, Chapter 5). We say that a function $\langle\cdot|\cdot\rangle$ that maps pairs of vectors $\mathbf{x}, \mathbf{y}$ in a vector space $\mathcal{V}$ to the nonnegative real numbers is an inner product, precisely when it satisfies, for all vectors $\mathbf{x}, \mathbf{y} \in \mathcal{V}$, the conditions

1. $\langle\mathbf{x}|\mathbf{x}\rangle \geq 0$, equality holding if and only if $\mathbf{x} = 0$
2. $\langle\mathbf{x}|\mathbf{y}\rangle = \langle\mathbf{y}|\mathbf{x}\rangle$,
3. $\langle\mathbf{x}|\alpha\mathbf{y} + \mathbf{z}\rangle = \alpha\langle\mathbf{x}|\mathbf{y}\rangle + \langle\mathbf{x}|\mathbf{z}\rangle$ for any scalar α

The reader easily verifies that the standard function $\mathbf{x}'\mathbf{y} = \sum_i \mathrm{x}_i\mathrm{y}_i$ indeed satisfies the above requirements. To open up the possibility to incorporate comparisons between unequally indexed coordinates, we first write the standard inner product, using the identity matrix $\mathbf{I}$ in a trivial way:

$$\mathbf{x}'\mathbf{y} = \mathbf{x}'\mathbf{I}\mathbf{y} = (\,\mathrm{x}_1, \quad \cdots \quad, \mathrm{x}_n\,)\begin{pmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{pmatrix}\begin{pmatrix} \mathrm{y}_1 \\ \vdots \\ \mathrm{y}_n \end{pmatrix}. \tag{15}$$

But this trivial extension invites to exchange $\mathbf{I}$ for the matrix $\mathbf{M}^*$ of matchings and calculate

$$\langle\mathbf{x}|\mathbf{y}\rangle = \mathbf{x}'\mathbf{M}^*\mathbf{y} = \sum_j \sum_i \mathrm{x}_i m_{ij} \mathrm{y}_j, \tag{16}$$

$$= \mathbf{x}'\mathbf{y} + \sum_{i \neq j} \mathrm{x}_i m_{ij} \mathrm{y}_j. \tag{17}$$

In Figure 2, we demonstrate that using an inner-product $\langle\mathbf{x}|\mathbf{y}\rangle = \mathbf{x}'\mathbf{M}\mathbf{y}$ with some nontrivial $\mathbf{M}$ will "distort" Euclidean distance through stretching or compressing the vector space in particular directions. The plots show how equidistance contours in $\{0,1\} \times \{0,1\}$ change as a result of changing $m_{12} = m_{21}$ in $\mathbf{M} = \begin{pmatrix} 1 & m_{12} \\ m_{21} & 1 \end{pmatrix}$, that is, in a vector space representing sequences defined over just two states.[4]

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

Just like actually constructing vectors from sequences is hardly practically feasible, it is not feasible either to generate the full matrix of matchings $\mathbf{M}^*$ since it has precisely as many rows and columns as $\mathbf{x}$ has coordinates: as many as there are nonnegative integers! Fortunately, a kernel function exists (Elzinga and Wang 2013) that allows for evaluating $\mathbf{x}'\mathbf{M}^*\mathbf{y}$ without actually constructing $\mathbf{M}^*$ or either of the vectors.

Spell Sequences: Handling Durations

Durations of Subsequences

At first sight, handling durations in the context of vector representations is easy to conceptualize. Let $T_x(u)$ denote the total duration of some subsequence $u \sqsubseteq x$. A representation $\mathbf{x} = (x_1 \ldots)$ of a sequence x is easily defined as

$$x_{r(u)} = \begin{cases} T_x(u) & \text{if } u \sqsubseteq x \\ 0 & \text{otherwise} \end{cases}, \tag{18}$$

leading to an inner product of the form $\mathbf{x}'\mathbf{y} = \sum_{u \in \mathcal{S}(x,y)} T_x(u) T_y(u)$. However, $T_x(u)$ may be ill-defined since when u has multiple embeddings in x, it is not evident how $T_x(u)$ should be measured. For example, consider $(x = abac; \mathbf{t}_x = [4, 3, 2, 3])$. Then, for $u = ac$, we have that the state a has two different durations and hence the duration $T(u)$ cannot be well defined. To properly deal with embeddings, we have to formally define the concept as follows (see also Elzinga et al. 2008; Elzinga and Wang 2013): We refine our notation through writing $i(u)$ to denote an embedding, that is, a sequence of position numbers that spells u in x. For example, for $x = abac$ and $u = bc$, we would have $i(u) = 2, 4$. For $u = ac$ we have $i(u) = 1, 4$ and $i'(u) = 3, 4$ and therefore we introduce the set $I_x(u)$ of all embeddings of u in x. When multiple embeddings do not occur, it is straightforward to define

$$T_x(u) = \sum_{j \in i(u)} t_j, \tag{19}$$

that is, as the sum of the lengths of the spells.

When multiple embeddings do exist, we could set $T_x(u)$ equal to the average of the durations of all embeddings:

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

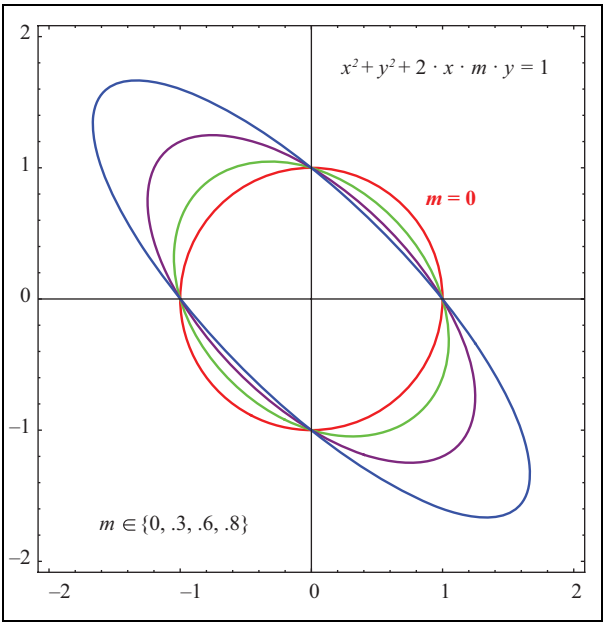

Figure 2. Unit distance plots in (x_1, x_2)-plane for various values of the coordinate matching-measure m_{12} of the elliptical inner product $\langle \mathbf{x}|\mathbf{x}\rangle = \mathbf{x}'\mathbf{M}^*\mathbf{x}$. The circle arises when $m_{12} = 0 = m$, that is, it represents the unit circle in "flat," standard inner product space. As m gets bigger and approaches 1, the circle is ever more elliptically deformed. For more than two dimensions, the unit sphere becomes an ellipsoid (not shown). *Note*. Color version of the figure is available online at smr.sagepub.com.

$$\overline{T}_x(u) = |x|_u^{-1} \sum_{i(u)\in I_x(u)} \sum_{j\in i(u)} t_j, \tag{20}$$

but this will rarely be an appealing option since it could imply mapping quite different sequences onto the same vector. Alternatively, we might use the durations of all embeddings. So, we define[5] the sum of all durations of all embeddings of a particular subsequence u as

$$T_x(u) = \sum_{i(u)\in I_x(u)} \sum_{j\in i(u)} t_j, \tag{21}$$

which can be interpreted as mapping embeddings, weighted for duration. The inner product resulting from this construction will then have the form

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

$$\mathbf{x}'\mathbf{y} = \sum_i x_i y_i \tag{22}$$

$$= \sum_{u \in \mathcal{S}(x,y)} \left(\sum_{i(u) \in I_x(u)} \sum_{j \in i(u)} t_{j;x} \right) \left(\sum_{i(u) \in I_y(u)} \sum_{j \in i(u)} t_{j;y} \right), \tag{23}$$

$$= \sum_{u \in \mathcal{S}(x,y)} \left(|x|_u \overline{T}_x(u) \right) \cdot \left(|y|_u \overline{T}_y(u) \right). \tag{24}$$

Equation (24) hints to an easy interpretation of the representation: Vector coordinates are averages of the durations of subsequences, weighted by their embedding frequencies.

Some authors (e.g., Abbott and Hrycak 1990; Halpin 2010) have suggested to transform time through a convex or concave function. This can be incorporated in Equation (23) by writing $f(t)$ instead of t.

Let us consider the function $f(t) = t^a$. If $a = 0$, no timing information will be used, and the algorithm is strictly equivalent to computing the distance between distinct states sequences. If $0 < a < 1$, then longer spells will weight comparatively less. Halpin (2010) has argued that this is an interesting feature. Finally, if $a > 1$, then small spells will be less important in sequences comparison. For instance, one may be interested in ignoring small unemployment episode while taking into account the longer ones.

Practical Considerations

All of the metrics discussed previously have been implemented in the freely[6] available software package TraMineR (Gabadinho et al. 2011) and the required algorithms have been amply described in Elzinga et al. (2008) and in Elzinga and Wang (2013); here we will not deal with algorithmic issues.

TraMineR imposes no practical limitations on the size of the alphabet or the number of sequences in the data set to analyze. However, with N sequences, the number of distinct pairs of sequences amounts to $\binom{N}{2} = N(N-1)/2$. This implies that the computation time for the distance matrix is roughly quadratic in the number of sequences: Doubling the size of the data set will lead to an almost fourfold amount of computation time required. For this reason, Studer (2013) proposed a procedure to analyze the data relying only on unique sequences by weighting them accordingly.

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

Let n denote the length of the sequences involved in a single distance computation; then the calculation of each inner product will be proportional to n^3. Hence, the total computation time involved in calculating the full distance matrix for a data set consisting of N sequences of length n will be roughly proportional to N^2n^3. A detailed analysis of the computational complexity of the algorithms involved can be found in Elzinga and Wang (2013). OM requires computation time proportional to only N^2n^2, but this difference is not very relevant as the following case illustrates.

McVicar and Anyadike-Danes (2002) published a data set consisting of 712 sequences of school-to-work transitions, each covering 72 months. Calculating a full distance matrix using TraMineR for this data set requires only 1.06 seconds for SVR (spell) while it requires 3.24 seconds for OM. The difference can be explained by the conversion from state sequences to spell sequences. As it greatly reduces the average length n of the sequences (decreasing from 72 to 3.5), computation time reduces by factor of $\frac{72^2}{3.5^3} = 121$. However, with fast modern PCs, these differences will be insignificant in most applications.

Converting equally long state sequences to spell sequences will normally generate spell sequences of unequal length. However, contrary to OM, working with vector representations does not require the sequences to be equally long. The reason is that the vector representing the shorter of the sequences will have zero-valued coordinates for all subsequences that are longer than the sequence itself. Therefore, multiplying vectors representing sequences of unequal length will only result in zero-valued products of coordinates referring to longer subsequences. Hence, there is no theoretical or practical objection whatsoever to calculating distances between sequences of unequal lengths.

The General Framework: Feature Vectors

So far, we presented several examples of a very general model for representing sequences as vectors, the coordinates indexed, given the state alphabet, by the sequences that are constructible from this alphabet. Given a sequence $x = x_1 \ldots x_n$, we constructed vectors $\mathbf{x} = (\mathrm{x}_1, \mathrm{x}_2, \ldots)$ such that the coordinate values are set to

$$\mathrm{x}_{r(u)} = \begin{cases} f(u,x) & \text{if } u \sqsubseteq x \\ 0 & \text{otherwise} \end{cases}, \tag{25}$$

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

wherein f is some function that maps the pair (u, x) to some number. In computer science, vectors that contain quantified information about discrete structures like graphs or strings are called "feature vectors" and sometimes the resulting vector space is called a "feature space." Here the set of features corresponds to the set of constructible sequences. In equation (25), the definition of the feature vectors is very general: The "weighting function" f may operate on both the subsequence indexed by r and on the sequence in which it is embedded. However, we have seen examples where f only operates on the subsequence u and not on x. We illustrate this in Table 2.

In the first entry of Table 2, $f(u, x) = 1$ whenever $u \sqsubseteq x$ and regardless of the features of u and regardless of the sequence x: The result is that the standard inner product $\mathbf{x}'\mathbf{y}$ equals the count of the number of distinct common subsequences. In the second entry of this table, the subsequences are weighted according to their length and hence f only operates on u: $f(u, x) = \sqrt{L(\ell(u))}$. In the third entry, the weighing depends on both x and u: $f(u, x) \neq f(u, y)$ precisely when $|x|_u \neq |y|_u$. The same is true for duration weighting: both $|x|_u$ and $\overline{T}_x(u)$ depend on both u and on x.

In the last two entries, we mention two kinds of weighting not dealt with in this article. Weighting according to gap width is relevant when one considers common subsequences with big time gaps between the states as less relevant. Weighting according to the state composition of the subsequences might be relevant when the occurrence of particular states is more salient than the occurrence of other states. The point here is that any kind of weighting can be accommodated in the general representation and it can be applied as long as we can find algorithms that allow us to evaluate the inner products of the vectors. Furthermore, it is important to stress the fact that any number of these weightings may be applied simultaneously, again provided suitable algorithms are available.

What is not shown in Table 2 is that each of these weightings can be applied with or without soft-matching of states, that is, with either an inner product of the form $\mathbf{x}'\mathbf{y}$ or of the form $\mathbf{x}'\mathbf{M}^*\mathbf{y}$ as long as $\mathbf{M}^*$ is positive semidefinite.

So, relying on a subsequence vector representation (SVR for short) allows for an enormous versatility in weighting features, warping time, applying soft-matching, and dealing with sequences of unequal lengths. Furthermore, the interpretation of the results of well-known methods in sequence analysis is made easier. For instance, using "Ward" clustering with such a metric is equivalent to finding clusters minimizing the residual variance of the features, that is, minimizing the variability of the subsequences. Using discrepancy analysis is equivalent to running a MANOVA in which the dependent variables are the features (i.e., the subsequences).

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

Table 2. Weighted Functions for Feature Vectors.

Feature Weighted	$f(u,x)$ if $u \sqsubseteq x$	Section
None	1	2.2
Length	$\sqrt{L(\ell(u))}$	2.4
Embedding frequency	$\lvert x\rvert_u$	2.3
Duration	$\lvert x\rvert_u \overline{T}_x(u)$	4.1
Gap width	(see Elzinga and Wang 2013)	—
State composition	(see Elzinga and Wang 2013)	—

Note. The middle column shows the evaluation of $f(u,x)$ for the kind of weighting as indicated in the leftmost column. The rightmost column shows where this kind of weighting is discussed in the main text. The last two kinds of weighting are not discussed in this article.

In the next two sections, we will compare SVR metrics with OM. In particular, we will use weighting of subsequence lengths by varying the parameter a in $L(\ell(u)) = \ell(u)^a$. If $a = 0$, no length weighting is applied and if $a > 0$, more weight is given to longer subsequences and the resulting SVR metric will be more sensitive to ordering. Furthermore, we will use time transforms of the form $f(t) = t^b$ previously introduced. The various metrics to be used are listed in Table 3.

Assessing Metric Sensitivity

Common *order* of states is the basic property that defines similarity between sequences as temporal successions of states or events (Elzinga 2003). However, common order is not the only angle from which to look at sequence similarity. Another important aspect is *duration*. For example, $a \ldots ab$ and $abbb \ldots b$ are quite different sequences, although the order in which a and b appear is the same. For instance, a difference in the duration of a poverty spell may have a huge impact on the rest of the life course, because poverty may act as a trap (Pollak 2010). Finally, *timing* of events can be the feature of interest as Lesnard (2010) and Rousset and Giret (2007) argued. For example, work during daytime is socially quite different from work during a night shift, and early unemployment may have quite a different effect than unemployment that occurs later in the career (Mooi-Reci 2012). Therefore, we will compare OM to different configurations of the newly introduced distance measures and evaluate how sensitive these measures are to differences in state order, in state duration and in state timing through using simulated, short sequences with controlled variations on these facets. A more detailed presentation of this simulation framework is available in Studer (2012).

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

Table 3. Metrics Used in Assessing Sensitivity; All of Them Weighted for Embedding Frequency.

Acronym	Description
NMS	NMS distance as defined in Elzinga (2003, 2005), that is, duration coded as replicated states and no subsequence length weighting. The present article extends this metric by allowing states proximities and durations
SVR (emb, spell, $a = 0$, $b = 1$)	Spell sequences, no subsequence length weighting, no time transform
SVR (emb, spell, $a = 1$, $b = 1$)	Spell sequences, subsequence length weighting using $L(\ell(u)) = \ell(u)$, no time transform
SVR (emb, spell, $a = 0$, $b = 2$)	Spell sequences, no subsequence length weighting, time transform using $f(t) = t^2$
OM	Standard OM algorithm, indel set as half the maximum substitution cost

Note. Emb = embedding frequency; OM = optimal matching; SVR = subsequence vector representation.

To evaluate these sensitivities, we proceed as follows. We generate two groups of sequences that differ in only one of the facets: in ordering, in timing or in duration. We then evaluate the ability of each distance measure to discriminate between these two groups using a Discrepancy Analysis. This analysis evaluates the strength of the association between the sequences as described by a distance measure and a partition (here, our two groups).[7] This association is measured using a pseudo-R^2 defined as

$$0 \leq R^2 = \frac{SS_B}{SS_T} \leq 1, \tag{26}$$

wherein SS_B and SS_T are sums of distances (for details, see Studer et al. 2011). Given a fixed set of sequences, the size of this R^2 will depend on the distance metric used, and the relative size of R^2 can thus be interpreted as a measure of how well a particular metric discriminates between groups of sequences. If this pseudo-R^2 is close to one, the distance measure is very sensitive to the facet on which the two groups differ. On the other hand, if the pseudo-R^2 is close to zero, the distance measure is relatively insensitive to the pertaining facet.

In order to get stable results, one million sequences were generated in each group of sequences. Each simulation is repeated one thousand times and the results proposed here show the average pseudo-R^2 over all runs.[8]

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

Confidence intervals are not plotted because standard errors are extremely small (maximum standard error of 1.9×10^{-4}).

In the present context, we ran three different types of simulations as summarized in Table 4. Each of these *separately* evaluates the sensitivity of the metric to perturbations of one of the facets previously introduced: ordering, timing, or duration. Subsequently, we present the details of each of these simulations and discuss the results.

Ordering and State Proximities

For each simulation, we created two groups of spell sequences: $(x, \mathbf{t}_x)$ and $(y, \mathbf{t}_y)$ with distinct but fixed patterns x and y. The duration vectors were randomly generated with the only restriction that the sum of the durations was fixed to 20 units of time, that is, $\mathbf{1}'\mathbf{t}_x = 20 = \mathbf{1}'\mathbf{t}_y$ for all sequences. For example, with $x = ca$ and $y = cb$, the simulated data set would look like

$$
\begin{aligned}
&[ca, (3, 17)]\\
&[ca, (12, 8)]\\
&\vdots\\
&[cb, (6, 14)]\\
&[cb, (15, 5)]\\
&\vdots
\end{aligned}
$$

A metric that is very sensitive to differences in order or pattern will easily separate the two groups by generating a high value of R^2, whereas a metric that is less sensitive to pattern will generate a lower R^2 because it will generate only a small distance between for example, $[ca, (19, 1)]$ and $[cb, (17, 3)]$ because the time spent in c is long in both sequences.

State proximities strongly affect the ordering. As a first example, consider again the spell sequences $x = (ca, \mathbf{t}_x)$ and $y = (cb, \mathbf{t}_y)$, all of length 20 and random durations. Now suppose that $m(a, b) = 1$, implying, in fact, that the states a and b are indistinguishable. As a result, x and y sequences cannot be separated, and hence we expect a discrepancy analysis to produce $R^2 = 0$. If, on the other hand, $m(a, b) = 0$, x and y sequences are easily separable, and we would expect R^2 to be close to 1.

Summarizing: we generate a set of sequences with two generators, calculate distances with one of the metrics from Table 3, calculate R^2 and repeat

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

Table 4. Patterns, Onset, and Duration Variations Used in Assessing the Sensitivity to Perturbations of Ordering, Timing and Duration. Total Duration of all Patterns Is Restricted to 20 Units of Time.

Simulation	Description	Group 1	Group 2
Ordering	Time spent in each state is random	*ca* *ac* *cac* *caca*	*cb* *cb* *cbc* *cbcb*
Timing	Random patterns *abc* or *cba*. *b* starts at time $2 + t$	$t = 0$	$t \in 2 \ldots 8$
Duration	Random patterns *abc* or *cba*. Duration of *b* equals $4 + t$	$t = 0$	$t \in 2 \ldots 8$

Note. Total duration of all patterns is restricted to 20 units of time.

this for different values of $m(a, b)$, and plot the values of R^2 against the values of $m(b, c)$. The results of these exercises are shown in Figure 3.[9]

The results when $m(a, b) = 0$ show the difference in sensitivity to ordering. SVR (spell) variants are the most sensitive to ordering. As expected, this sensitivity increases with subsequence length weighting ($a = 1$), and it decreases when spell durations are squared ($b = 2$), the latter having a stronger effect. OM is much less sensitive to ordering, and NMS (SVR with replicated states) fails to identify a difference in the orderings. Similar results pertaining to OM and NMS were already discussed in Studer (2012). Since we are measuring *relative* sensitivity, the distance measure could be more sensitive to other facets of sequence comparisons.

The shapes of the curves convincingly show the effect of soft-matching. In a qualitative sense, the SVR metrics show the same behavior as the OM metric and they all behave as expected. In all cases, the R^2 are maximal when $m(a, b) = 0$ and the R^2 tends toward zero when $m(a, b) = 1$. SVR (spell) variants react more pronounced than OM, because matching subsequences are proportionally weighted by their durations. On the contrary, NMS underperforms, because the number of matching subsequences $\mathbf{x}'\mathbf{y}$ is always small relative to the quantities $\mathbf{x}'\mathbf{x}$ and $\mathbf{y}'\mathbf{y}$.

Timing

Timing simulations follow the same logic as the one for ordering. Patterns and durations are random in both groups, but the spell in the state *b* always

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

starts at time $2 + t$. We set $t = 0$ in the first group and progressively change t in the second one. Here again, we are measuring the *relative* sensitivity of the metrics to timing.

The first panel of Figure 4 presents the evolution of the R^2 when the time difference between both groups increases for each of these simulations. NMS is the most sensitive to timing, but the slope of the curve decreases, meaning that it comparatively fails to discriminate between very high difference of timing. SVR (spell, $a = 0$, $b = 2$) performs very well while the other shows very similar results.

Duration

We used the same strategy again for the duration simulations. Patterns and timing are random, while the duration of the second spell (in the state b) is set to 4 in the first group. In the second group, this duration is set to $4 + t$ and t is progressively changed in the interval $t \in 2 \ldots 8$. Here again, we are measuring the *relative* sensitivity to duration of the metrics, since the metric may still be more sensitive to other facets (such as ordering).

The second panel of Figure 4 presents the results for duration-related simulations. By far, OM is the most sensitive to duration. NMS and SVR (spell, $a = 0$, $b = 2$) present, here again, an intermediary position. Regarding SVR (spell), we note that b parameter is strongly linked with sensitivity to timing and duration. The a parameter lowers the sensitivity too, but the effect is not very pronounced.

Conclusion

According to our simulations, the distance measures are sensitive to different facets. SVR (spell) variants are most sensitive to differences in ordering, NMS is most sensitive to timing and OM to duration. This means that the choice of a distance measure always has to be justified in the context of the application in which it is applied.

These simulations allowed us to measure the effects of the SVR (spell) parameters and to demonstrate that they behave as expected. The a parameter raises the sensitivity to ordering and lowers sensitivity to timing and duration. On the contrary, raising b leads to a distance measure that is more sensitive to timing and duration and less to ordering.

We now turn to an application of these SVR metrics to real data in order to highlight the contributions of the newly introduced distance measures.

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

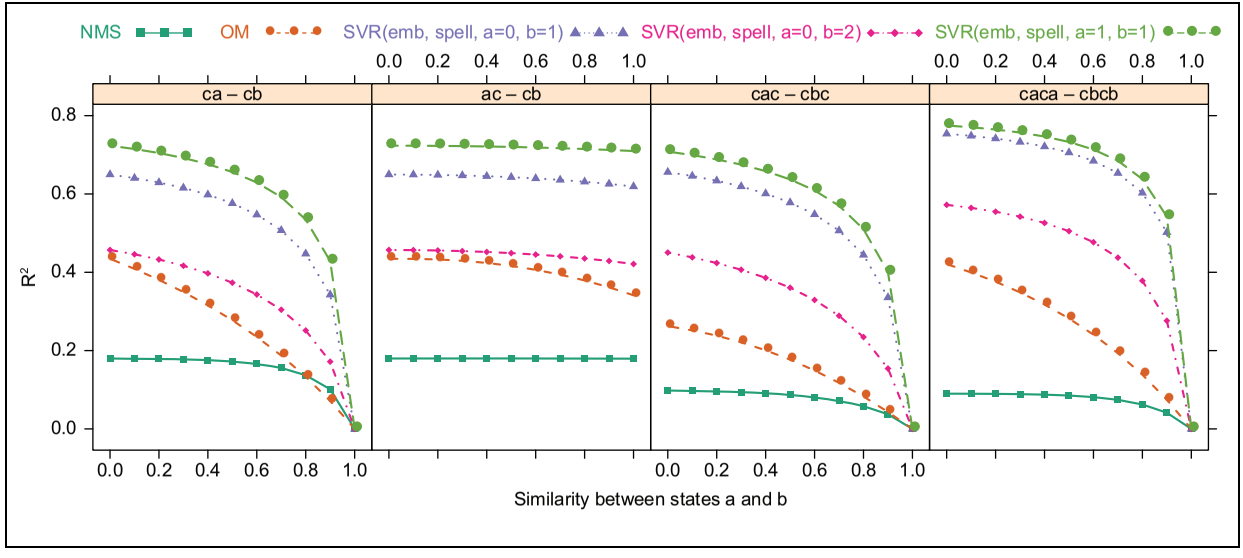

Figure 3. Plots of discrepancy analysis' R^2 (vertical axis) vs. $0 \leq m(a, b) \leq 1$ (horizontal axis) for the metrics of Table 3. The plots result from analyzing the pairs of spell sequences constructed from the patterns *ca* – *cb*, *ac* – *cb cac* – *cbc*, and *caca* – *cbcb* (right panel).

Note. Color version of the figure is available online at smr.sagepub.com.

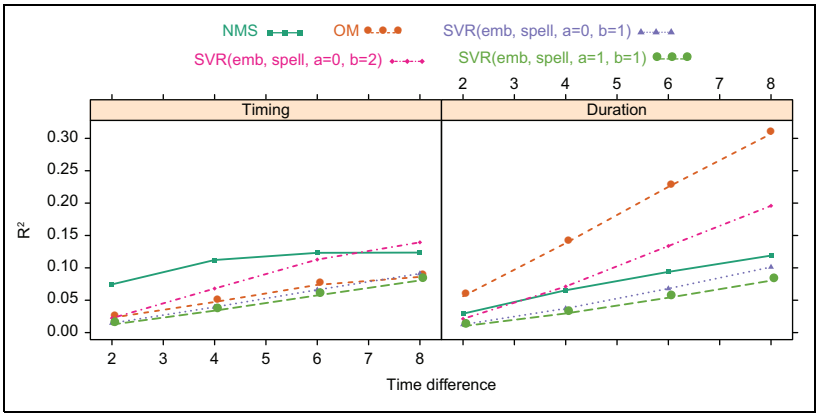

Figure 4. Evolution of the R^2 (vertical axis) while varying time difference (horizontal axis) for different metrics (different lines), resulting from the discrepancy analysis of the time related simulation summarized in Table 4.
Note. Color version of the figure is available online at smr.sagepub.com.

An Application to Family Formation

Data and Distances

In this section, we apply the different configurations of the SVR metrics to well-known data and compare the results with those obtained when applying OM and NMS to the same data. The data were first presented in Müller et al. (2008).

Briefly, these data represent family formation trajectories of Swiss individuals who were at least 30 years old at the time of the survey.[10] One of the goals of this study was to highlight the change of the social norms constraining these trajectories. The states in the sequences were built using a combination of four distinct events: Leaving home, Marriage, having a first Child, and Divorce. For the sake of simplicity, some very rare states were merged resulting in eight possible states. An individual is in the state "P" (living with Parent) if no event has occurred, in the state "L" if the event "Left parental home" occurred, in the state "LM" for "Left and Married," and "LMC" for "Left, Married and with a first Child." Similarly, state "M" is for an individual who just Married (without leaving parental home), and so on. Finally, state "D" is for all individuals who have married and divorced (without making difference for having left the parental home and/or having or not having children).

To determine the substitution costs needed for the calculation of an OM distance matrix, we proceeded as follows. First, we created a four-dimensional

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

Table 5. State Definitions of Family Formation Trajectories From the Swiss Household Panel.

	Events			
States	Leaving home	First marriage	First child	First divorce
P	No	No	No	No
L	Yes	No	No	No
M	No	Yes	Yes/No	No
LM	Yes	Yes	No	No
C	No	No	Yes	No
LC	Yes	No	Yes	No
LMC	Yes	Yes	Yes	No
D	Yes/No	Yes	Yes/No	Yes

Note. C = child; D = individuals who have married and divorced; L = Left parental home; P = living with Parent; LM = left and married; LMC = left, married and with a first child; NMS = number of matching subsequences; sp = spell; SVR = subsequence vector representation.

vector for each state; the coordinates corresponding to the events shown in Table 5 by assigning 0 to "no," 1 to "yes," and 0.5 to "yes/no." Then, we calculated the Manhattan distance between all pairs of states and normalized these distances to the maximum distance found (= 3). This amounts to assigning a value of $\frac{1}{3}$ to each coordinate distance. These costs can thus be interpreted as the difference in the events that already happened. For example, the substitution costs between states "P" having coordinates $(0, 0, 0, 0)$ and "LM" with coordinates $(1, 1, 0, 0)$ equals $\frac{|1-0|+|1-0|+|0-0|+|0-0|}{3} = \frac{2}{3}$. Since the SVR-metric needs *proximities* instead of *costs*, we set the soft-matching coefficients to 1 minus the OM cost. Taking the same example, the proximity between states "P" and "LM" is thus $1 - \frac{2}{3} = \frac{1}{3}$.

In order to compare the results obtained by using different metrics, we calculated the different distance matrices using the proximities (or costs) defined previously: SVR (sp, $b = 1$; SVR based on spells), SVR (sp, $b = 2$; SVR based on spells, squared durations), SVR (sp, $a = 1$; SVR based on spells with subsequence weighting), and the OM distance. In order to highlight the effect of proximities, we also added the distance SVR (sp, $b = 1$, c), the SVR (sp, $b = 1$) distance computed using constant differences (i.e., a similarity of zero between all states). Finally, we included the NMS distance as defined by Elzinga, that is, with constant cost, in order to highlight the distinctive features of the newly proposed metrics. These are the

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

same metrics as used in the simulations (see Table 3), but with shortened names in order to generate useful plots.

Distance Disagreements

To investigate the differences between the various metrics, we started by looking for pairs of sequences where the different metrics generate very different distances. Thereto, we first standardized the metrics in order to get rid of different distance units; for each metric, say metric a, we divided all distances by the standard deviation of the distances as generated by that metric. Thus, we calculated, for all pairs of sequences,

$$d'_a(x,y) = \frac{d_a(x,y)}{s_{d_a}},$$

with the effect of creating dimensionless or unit-free distances d'_a that can be compared across metrics.[11] Next, for two metrics, say a and b, we looked at the pair of sequences (x,y) for which the difference $d'_a(x,y) - d'_b(x,y) = \Delta(a,b)$ is maximal. Since $\Delta(a,b)$ may be negative, we also looked for pairs of sequences (x',y') for which $\Delta(b,a) = d'_b(x',y') - d'_a(x',y')$ is maximal. This procedure generates a matrix of pairs of sequences and values of Δ as shown in Table 6. The Δ is computed by subtracting the distance given in column to the one given in the row. Thus, in each cell, we find pairs of sequences for which the pertaining metrics generate extremely different distances.

Let us discuss an example by looking at the strongest disagreement between standardized OM and standardized SVR (sp, $b = 1$). In the first column fourth row, we have "OM – SVR (sp, $b = 1$) = 4.74" for the comparison of the sequences $P^{15} - LMC^1$ and $P^2 - LMC^{14}$. According to OM, these sequences are far away because OM is strongly related to the total time spent in each state, which are very different in this case. According to SVR (sp, $b = 1$), these sequences are close, because SVR (sp, $b = 1$) is more linked to the order of the states, which is the same in both sequences. We can also have a look at the reverse, that is when SVR (sp, $b = 1$) is greater than the OM distance. This is found when comparing the sequence $P^1 - L^5 - LM^1 - LMC^8 - D^1$ and $L^8 - LMC^8$. According to SVR (sp, $b = 1$), these sequences are far away, because the ordering of the states is different. According to OM, the sequences are close because the time spent in states L and LMC are more or less the same.

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

Table 6. Analysis of Biggest Standardized Distance Differences.

	SVR (sp, $b = 1$)	SVR (sp, $b = 2$)	SVR (sp, $a = 1$)	OM	SVR (sp, $b = 1$, c)	NMS
SVR (sp, $b = 1$)		$\Delta = 2.21$ L ^{9}-LMC ^{7} P ^{3}-L ^{4}-LM ^{3}-LMC ^{4}-D ^{2}	$\Delta = 0.63$ L ^{15}-LC ^{1} P ^{2}-M ^{14}	$\Delta = 4.67$ L ^{7}-LMC ^{9} P ^{1}-L ^{5}-LM ^{1}-LMC ^{8}-D ^{1}	$\Delta = 0.9$ P ^{5}-L ^{2}-LM ^{1}-LMC ^{3}-D ^{5} P ^{16}	$\Delta = 4.03$ P ^{4}-L ^{3}-LC ^{5}-LMC ^{3}-D ^{1} P ^{8}-M ^{2}-LM ^{1}-LMC ^{1}-D ^{4}
SVR (sp, $b = 2$)	$\Delta = 3.69$ P ^{1}-L ^{13}-LM ^{1}-LMC ^{1} P ^{4}-L ^{4}-LM ^{4}-LMC ^{4}		$\Delta = 3.98$ P ^{1}-L ^{13}-LM ^{1}-LMC ^{1} P ^{2}-M ^{14}	$\Delta = 6.2$ L ^{16} P ^{1}-L ^{13}-LM ^{1}-LMC ^{1}	$\Delta = 3.13$ P ^{1}-L ^{13}-LM ^{1}-LMC ^{1} P ^{16}	$\Delta = 4.29$ P ^{1}-L ^{13}-LM ^{1}-LMC ^{1} P ^{8}-M ^{2}-LM ^{1}-LMC ^{1}-D ^{4}
SVR (sp, $a = 1$)	$\Delta = 0.21$ P ^{7}-L ^{1}-LM ^{1}-LMC ^{3}-D ^{4} P ^{8}-M ^{2}-LM ^{1}-LMC ^{1}-D ^{4}	$\Delta = 2.24$ P ^{3}-L ^{4}-LM ^{3}-LMC ^{4}-D ^{2} P ^{4}-L ^{6}-LMC ^{6}		$\Delta = 4.68$ L ^{7}-LMC ^{9} P ^{1}-L ^{5}-LM ^{1}-LMC ^{8}-D ^{1}	$\Delta = 0.78$ P ^{5}-L ^{2}-LM ^{1}-LMC ^{3}-D ^{5} P ^{16}	$\Delta = 4.09$ P ^{4}-L ^{3}-LC ^{5}-LMC ^{3}-D ^{1} P ^{8}-M ^{2}-LM ^{1}-LMC ^{1}-D ^{4}
OM	$\Delta = 4.74$ P ^{2}-LMC ^{14} P ^{15}-LMC ^{1}	$\Delta = 2.7$ P ^{3}-LM ^{6}-D ^{7} P ^{5}-C ^{11}	$\Delta = 5.31$ P ^{2}-LMC ^{14} P ^{16}		$\Delta = 4.24$ P ^{2}-LMC ^{14} P ^{15}-LMC ^{1}	$\Delta = 2.4$ P ^{1}-L ^{3}-LC ^{10}-LMC^{2} P ^{4}-M ^{2}-LM ^{1}-D ^{9}
SVR (sp, $b = 1$, c)	$\Delta = 2.27$ P ^{2}-L ^{7}-LMC ^{6}-D ^{1} P ^{7}-LM ^{4}-LMC ^{3}-D ^{2}	$\Delta = 2.91$ P ^{3}-L ^{4}-LM ^{3}-LMC ^{4}-D ^{2} P ^{4}-L ^{3}-LC ^{5}-LMC ^{3}-D ^{1}	$\Delta = 2.46$ P ^{2}-L ^{1}-LMC ^{11}-D ^{2} P ^{10}-LM ^{4}-LMC ^{1}-D ^{1}	$\Delta = 4.61$ P ^{1}-L ^{5}-LM ^{1}-LMC ^{8}-D ^{1} P ^{2}-L ^{5}-LC ^{2}-LMC ^{6}-D ^{1}		$\Delta = 4.51$ P ^{3}-L ^{7}-LC ^{1}-LMC ^{2}-D ^{3} P ^{8}-M ^{2}-LM ^{1}-LMC ^{1}-D ^{4}
NMS	$\Delta = 9.99$ L ^{16} P ^{16}	$\Delta = 8.05$ L ^{16} P ^{16}	$\Delta = 10.49$ L ^{16} P ^{16}	$\Delta = 8.67$ L ^{16} P ^{16}	$\Delta = 8.99$ L ^{16} P ^{16}	

Note. D = individuals who have married and divorced; L = Left parental home; P = living with Parent; LM = left and married; LMC = left, married and with a first child; NMS = number of matching subsequences; sp = spell; SVR = subsequence vector representation. Table 6 specifies the distances and the pertaining sequence pairs.

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

We can identify the contribution of soft matching coefficients by looking for the differences between SVR (sp, $b = 1$) and SVR (sp, $b = 1, c$) (SVR (sp, $b = 1$) with or without soft matching coefficients). Using states proximities, the distance between $P^2 - L^7 - LMC^6 - D^1$ and $P^7 - LM^4 - LMC^3 - D^2$ is lower than using constant proximities, because states L and LM are close according to our soft matching coefficients. Accounting for states proximities allows to consider that sequences $P^5 - L^2 - LM^1 - LMC^3 - D^5$ and P^{16} are comparatively more distant. This is exactly what soft matching coefficients are intended to do.

We can also identify the contribution of SVR distances parameters such as time transform by looking at the differences between SVR (sp, $b = 1$) and SVR (sp, $b = 2$). As expected and confirming our simulation results, SVR (sp, $b = 2$) is more sensitive to time spent in each state whereas SVR (sp, $b = 1$) is more sensitive to ordering. Subsequences length weighting SVR (sp, $a = 1$) has the effect of weighting the comparison of states in sequences containing many different spells. As a results, $P^7 - L^1 - LM^1 - LMC^3 - D^4$ and $P^8 - M^2 - LM^1 - LMC^1 - D^4$ are considered to be farthest by SVR (sp, $a = 1$). On the contrary, SVR (sp, $b = 1$) is comparatively more sensitive to difference in short spell sequences ($L^{15} - LC^1$ and $P^2 - M^{14}$). However, in both cases, the differences are small (less than 1).

Finally, let us look at the difference between SVR metric and NMS (Elzinga 2003). Since NMS only accepts constant state proximities, we will compare distances SVR (sp, $b = 1$, c) and NMS.[12] According to NMS, sequences L^{16} and P^{16} are farthest, because NMS will count many different subsequences while SVR (sp, $b = 1, c$) will only consider one subsequences in each sequences. On the contrary, SVR (sp, $b = 1, c$) is comparatively more sensitive to difference of ordering in long spell sequences ($P^8 - M^2 - LM^1 - LMC^1 - D^4$ and $P^3 - L^7 - LC^1 - LMC^2 - D^3$).

The analysis of distance disagreement confirms the results of the simulations. SVR (sp) variants are the most sensitive to ordering while OM distance is strongly linked with the time spent in a state. This analysis also highlights more precisely the effect of the SVR parameters. While b increases the sensitivity to duration and timing, a makes the distance measure more sensitive to the ordering of *complex* sequences. Finally, this analysis has confirmed that soft matching has the desired effect. It highlights the main judgment differences between distances measures. However, in practice, all differences, even the smallest ones, are taken into account. We slightly varied the state proximities as discussed in this article and found no results that were unexpected; we do not report on these results since they are too limited to warrant firm conclusions on the sensitivity of the methods to small perturbations of

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

Table 7. Clustering Quality Measured Through Average Silhoutte Width (ASW, to be maximized, Kaufman and Rousseeuw [1990]) and the HC index (HC, to be minimized, Hubert and Levin (1976)) with Various Metrics as Indicated Subsequently.

Metric	*nc*	ASW	HC
SVR (sp, $b = 1$)	11	.55	.05
SVR (sp, $b = 1$, c)	10	.53	.08
SVR (sp, $b = 2$)	10	.42	.09
SVR (sp, $a = 1$)	12	.65	.02
OM	6	.37	.07
NMS	17	.35	.07

Note. ASW = Average Silhoutte Width; HC = Hubert's C; NMS; OM = ; SVR = subsequence vector representation; denote the optimal number of clusters for each of the metrics used.

the state proximities. However, a more thorough investigation of the sensitivity issue would unduly elongate this already length article.

Clustering

We used the partitioning around medoids (PAM) algorithm (Kaufman and Rousseeuw 1990) to cluster the sequences on the basis of the four distance matrices using the sampling weights. To get an indication of the optimal number of clusters, we calculated all solutions with the number of clusters varying between 3 and 20. Table 7 summarizes the results. Both the ASW and the HC index are dimensionless measures, each depending on a ratio (of differences) of distances, and therefore, these indices can be used to compare partitions based upon different distance matrices. They can be interpreted as the capacity of a clustering method to match the structure of the data, the structure being defined by the features of each metric. The computations were carried out with the WeightedCluster library (Studer 2013). SVR-based clusterings usually identify more clusters and the best clustering quality is found with SVR (sp, $a = 1$).

Table 8 presents the medoids of the clusters obtained using this optimal number of groups. SVR-metrics provide very similar clustering (Cramer's $V \geq 0.89$ between these solutions). SVR (sp, $b = 1$) and SVR (sp, $a = 1$) identify two small groups of trajectories leading to divorce that are not identified with other distances. SVR (sp, $a = 1$) also finds a small group of nonmarried parents (ending in state *LC*). These are important features, since all of these patterns may have gained in importance during the 20th century. If divorce is negligible, it should not be used to build the sequences. Otherwise, it should be included in subsequent analyses. SVR (sp, $b = 2$) makes some

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

distinctions between sequences according to the time spend in each state of the pattern $P - L$.

Confirming the sensitivity to timing highlighted by the simulations, NMS makes several distinctions according to the timing of transitions. However, confirming the results presented by Aisenbrey and Fasang (2010), all complex sequences are regrouped in a big, quite heterogeneous "residual" cluster ($P^3 - L^4 - LM^3 - LMC^4 - D^2$ that contains 37% of the sequences).

Next, we closely scrutinize the difference between the clustering results for OM and SVR (sp, $b = 1$) through visually rendering the clusters.

Cluster Visualization

To visually render the clusters, we will use parallel coordinate plots, chronograms, and sequence index plots. As many readers may not be familiar with parallel coordinate plots (for short: PC plots), we first spend a few lines on them (see also Bürgin and Ritschard 2014; Bürgin, Ritschard, and Rousseaux 2012; Inselberg 2009).

A PC plot renders multivariate objects on a flat surface by first drawing as many vertical lines as there are variables or dimensions, each of which may have a different scale. Individual objects are depicted as a line, drawn in left-to-right direction, crossing the vertical (parallel) lines at the appropriate height. Often, the thickness of the object-representing lines is proportional to the number of objects that share the same coordinates. A toy example of a PC plot is shown in Figure 5.

Here, we use the PC plots to render the sequences by the order of the events, ignoring durations. To attain this, we use as many identical, parallel scales, as there are events (states) in the individual sequences. Hence, an individual's position on the first of the scales corresponds to the first event, her position on the second parallel scale corresponds to the second event, and so on.

Figure 6 presents the PC plots of the sequences plotted according to the SVR (sp, $b = 1$) clustering. Let us discuss some examples in order to illustrate the interpretation of these plots. In the plot called "P-LM," the brown line indicates one of the patterns of the four events. It starts at position 1 in state "P" (living with parents) before going to the events "left parental home" and "marriage" at position 2. Since "left parental home" and "marriage" happen simultaneously, the line is vertical. In the group called "P-L-LM," the green line indicates that the pattern is "P," leaving at position 2 and marrying later on position 3. In both plots, the size of the squares and the width of the lines are plotted according to the relative frequency of the pattern.

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

Table 8. Medoids of the Clusters Found With Different Distance Metrics With Varying State Similarities.

SVR (sp, $b = 1$)	%	SVR (sp, $b = 2$)	%	SVR (sp, $a = 1$)	%	OM	%	SVR (sp, $b = 1$, c)	%	NMS	%
P^{16}	9.4	P^{16}	8.9	P^{16}	9.4	P^{16}	19.0	P^{16}	9.7	P^{16}	9.0
P^{9}-M^{7}	9.7	P^{9}-M^{7}	10.3	P^{9}-M^{7}	9.7	P^{8}-M^{8}	8.0	P^{9}-M^{7}	10.0	P^{13}-M^{3}	2.5
										P^{9}-M^{7}	4.0
										P^{6}-M^{10}	2.7
P^{7}-L^{9}	8.8	P^{10}-L^{6}	8.3	P^{7}-L^{9}	18.7	P^{5}-L^{11}	21.9	P^{10}-L^{6}	8.5	P^{13}-L^{3}	2.6
		P^{5}-L^{11}	11.5					P^{5}-L^{11}	11.9	P^{10}-L^{6}	4.2
										P^{7}-L^{9}	5.5
										P^{5}-L^{11}	5.9
										P^{1}-L^{15}	2.2
P^{10}-LM^{6}	7.1	P^{10}-LM^{6}	7.8	P^{10}-LM^{6}	7.2			P^{10}-LM^{6}	7.8	P^{14}-LM^{2}	1.9
										P^{10}-LM^{6}	2.7
										P^{7}-LM^{9}	2.4
P^{8}-LMC^{8}	7.9	P^{8}-LMC^{8}	7.9	P^{8}-LMC^{8}	8.0			P^{8}-LMC^{8}	8.5	P^{11}-LMC^{5}	2.8
										P^{7}-LMC^{9}	4.3
P^{7}-LM^{3}-D^{6}	1.9			P^{7}-LM^{3}-D^{6}	2.0						
				P^{6}-L^{5}-LC^{5}	1.1						
P^{6}-L^{6}-LM^{4}	10.2	P^{7}-L^{7}-LM^{2}	10.8	P^{6}-L^{6}-LM^{4}	10.2	P^{6}-L^{4}-LM^{6}	16.3	P^{6}-L^{6}-LM^{4}	11.0		
P^{6}-L^{5}-LMC^{5}	8.6	P^{4}-L^{6}-LMC^{6}	8.3	P^{5}-L^{5}-LMC^{6}	7.6			P^{5}-L^{5}-LMC^{6}	7.3		
P^{9}-LM^{2}-LMC^{5}	11.6	P^{9}-LM^{1}-LMC^{6}	12.2	P^{9}-LM^{2}-LMC^{5}	11.5			P^{9}-LM^{2}-LMC^{5}	11.6	P^{11}-LM^{2}-LMC^{3}	4.9
										P^{8}-LM^{1}-LMC^{7}	5.3
P^{5}-L^{3}-LM^{4}-D^{4}	1.1			P^{5}-L^{3}-LM^{4}-D^{4}	1.1						
P^{6}-L^{4}-LM^{2}-LMC^{4}	13.7	P^{6}-L^{5}-LM^{1}-LMC^{4}	14.0	P^{6}-L^{4}-LM^{2}-LMC^{4}	13.6	P^{10}-L^{1}-LM^{1}-LMC^{4}	17.1	P^{6}-L^{4}-LM^{2}-LMC^{4}	13.7		
						P^{5}-L^{2}-LM^{1}-LMC^{8}	17.6				
										P^{3}-L^{4}-LM^{3}-LMC^{4}-D^{2}	37.0

Note. D = individuals who have married and divorced; L = Left parental home; P = living with Parent; LM = left and married; LMC = left, married, and with a first child; sp = spell; SVR = subsequence vector representation. Durations of states are indicated as superscripts of the state acronyms. Clusters have been placed on the same row when their state-orders match. Relative cluster sizes ($N = 4,191$) are shown in the column labeled "%."

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

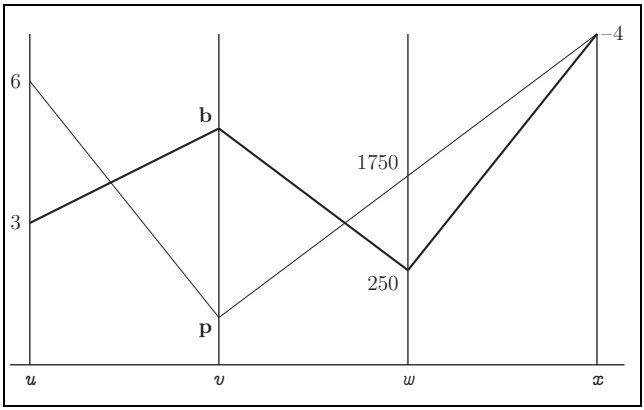

Figure 5. Parallel-coordinate plot of a multivariate object $(u, v, w, x) = (6, \mathbf{p}, 1750, -4)$ and 4 multivariate objects $(u, v, w, x) = (3, \mathbf{b}, 250, -4)$.

Using these plots, we can see that the SVR (sp, $b = 1$) clustering is very homogenous according to the ordering of the underlying events. Only the clusters leading to divorce group different patterns, but they all end with divorce. The clusters distinguish the sequences according to the synchronization of events, notably marriage and leaving home. In a first set of clusters, leaving home is experienced before marriage, while in another set these events occur simultaneously. These are important distinctions; Billari, Philipov, and Baizán (2001) argued that the simultaneity of marriage and leaving home should be interpreted as one distinct state.

Figure 7 presents the chronograms of the six clusters found from the OM distances. From these chronograms, the clusters seem easy to interpret. Indeed they are, but only on the basis of the time spend in the states and not on the basis of the orderings of the underlying events. Figure 8 presents the PC plots of the same clustering. The underlying orderings of the events are very diverse in each cluster. For instance, looking at the cluster called "Late LMC," at least four patterns can be identified (events in parenthesis occurs simultaneously): P-(LM)-C (in rose), P-(LMC) (dark-blue), (P-L-M-C (yellow), and P-L-(MC) (green). Using the chronogram, we are tempted to call the first group "Staying with parents," because the mean time spent in state "P" is large. However, the PC-plots show that many distinct patterns are grouped here.

The wide use of chronograms and index plots may be one of the reasons of the popularity of OM. As we have shown with our simulations and through

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

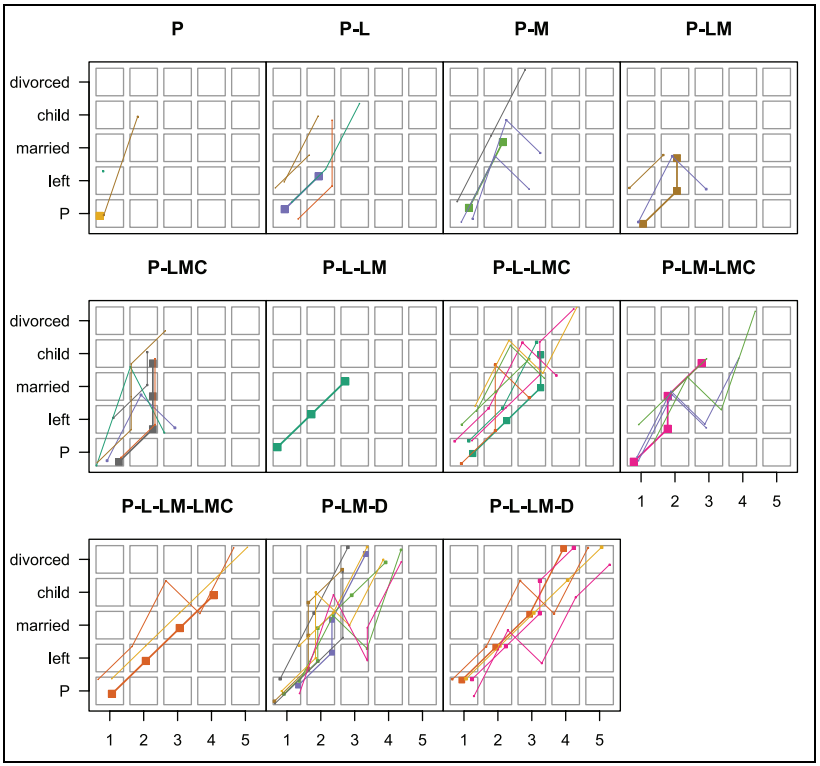

Figure 6. Parallel-coordinate plots of the 11 clusters found from the SVR (sp, $b = 1$)-distances.
Note. Color version of the figure is available online at smr.sagepub.com.

this example, OM is strongly linked with duration differences. This is shown in chronograms and index plots too, because the area plotted in single color depends of the total time spent in the associated state.

Comparing both clustering solutions, using OM leads to some distinctions according to time spent in each state while SVR (sp, $b = 1$) is strongly linked with the ordering of the underlying events.

Metrics and the Evolution of family trajectories

If there would be an evolution of family trajectories, we would expect to see the size of clusters change over time in a systematic way (see, e.g., Elzinga

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

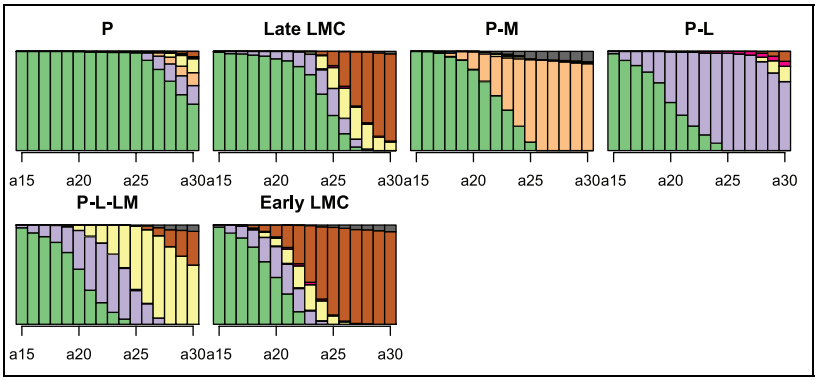

Figure 7. Chronograms of the six clusters found from the OM distances.
Note. Color version of the figure is available online at smr.sagepub.com.

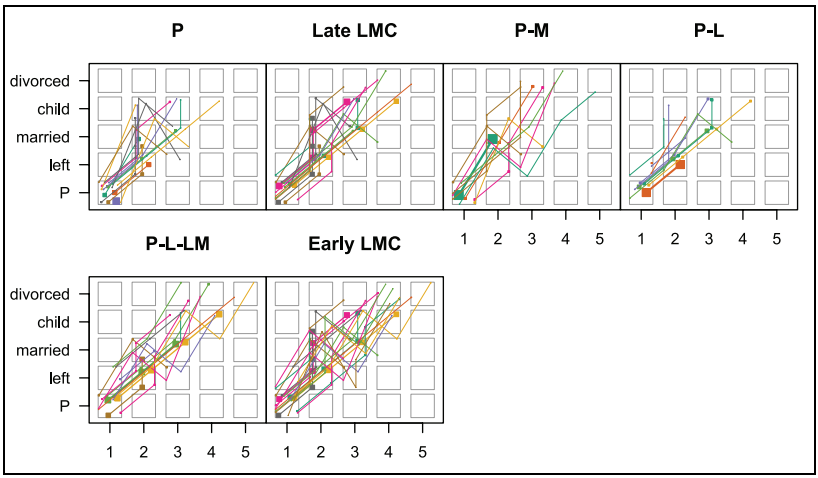

Figure 8. Parallel coordinate plots of the six clusters found from the OM distances.
Note. Color version of the figure is available online at smr.sagepub.com.

and Liefbroer 2007). Here, we evaluate these changes as revealed by both clustering on the basis of OM distances as well as on the basis of SVR (sp, $b = 1$). In Tables 9 and 10, we present the relative distributions of cluster membership per cohort, for OM- and SVR distances respectively. The association is highly significant in both cases but stronger for SVR-based clustering (Cramer's $V = .193$) than for OM-based clustering (Cramer's $V = .147$).

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

Table 9. Distributions of Relative Cluster Frequencies (%'s) per Cohort for OM-Based Clusters.

	< 30	$30 - 39$	$40 - 49$	$50 - 59$	≥ 60
P-M	13.4	12.8	10.7	5.4	3.8
P	35.5	21.1	15.0	13.8	18.9
Late LMC	17.7	23.0	18.7	16.0	13.6
P-L-LM	13.6	15.8	19.5	17.7	14.5
Early LMC	9.1	17.1	23.6	20.0	15.1
P-L	10.7	10.3	12.5	27.1	34.2

Note. Cells are colored in blue if the standardized Pearson residuals is higher than 1.96 and in red if lower than –1.96. Clusters are characterized by their medoids. Cramer's $V = 147$. Color version of the table is available online at smr.sagepub.com.

More interesting is the question if and what qualitative differences show up when we study the evolution of cohorts through OM or through an SVR-based metric.

Using OM, the evolution seems to be dominated by state duration changes. Older cohorts were staying longer with their parents (clusters $P - M$, P and Late LMC) while younger cohorts leave the parental home earlier. Moreover, the last cohort seems to distinguish itself by not marrying and not having children. However, caution is needed, because of the heterogeneity of the orderings.

Clustering SVR distances provides for an alternative view on this evolution by highlighting changes in the ordering of the events. Older cohorts stand out through the synchronicity of leaving the parental home and marriage. These two events were frequently occurring simultaneously, but this is much less frequent in the youngest cohorts. This "de-synchronization" has been interpreted as the result of the raise of nonmarital union in Switzerland and the introduction of a new intermediary stage of "partial independence" in the road toward autonomy (Thomsin et al. 2004). Contrary to OM, here the latest cohort does not distinguish by not marrying nor having children, but by different patterns leading to these situations.

Clearly, in this analysis of Swiss family formation sequences, the SVR (sp, $b = 1$)-based metric has provided new insights through revealing the underlying ordering of the events. OM leads to interesting results when we are interested in the durations spent in each state.

Conclusion and Discussion

We motivated this article by pointing at the poor performance of the OM metric with respect to a basic property of sequences: the order of the states

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

Table 10. Distributions of Relative Cluster Frequencies (%'s) per Cohort for SVR (sp, $b = 1$)-Based Clusters.

	< 30	$30 - 39$	$40 - 49$	$50 - 59$	≥ 60
P-M	19.4	15.9	12.0	5.6	4.6
P-LM	11.7	11.4	10.8	5.1	2.5
P-LM-LMC	12.8	17.2	14.4	11.9	6.1
P-LMC	11.6	10.8	10.2	7.8	3.9
P	19.1	10.1	8.3	7.1	8.3
P-LM-D	2.2	1.5	2.9	2.0	1.5
P-L-LM-D	0.2	0.4	0.7	1.2	1.8
P-L-LMC	2.6	6.4	8.3	9.7	11.2
P-L-LM-LMC	4.4	11.4	13.2	15.2	17.0
P-L-LM	5.8	6.6	8.6	12.8	12.8
P-L	10.2	8.3	10.7	21.5	30.4

Note. SVR = subsequence vector representation. Cells are colored in blue if the standardized pearson residuals is higher than 1.96 and in red if lower than −1.96. Clusters are characterized by their medoids. Cramer's $V = .193$. Color version of the table is available online at smr.sagepub.com.

or events involved. OM is not very sensitive to differences in the sequencing of the pertaining states. This lack of sensitivity is nicely demonstrated through the application described in the previous section through the PC plots that show very different orderings of the underlying events within clusters. This is not to say that OM cannot be a useful metric: It is useful when state durations are more important than state ordering. This too is shown in the chronograms of the previous section.

According to our simulations, the NMS-based metric is mostly sensitive to differences in timing. However, in the application presented, one of the big NMS cluster regroups all "complex" sequences which is not very meaningful. Such a phenomenon was already noted by Aisenbrey and Fasang (2010).

We presented a very flexible, versatile metric that does well when ordering of states is the key issue. This too was demonstrated in the previous section and in the simulations presented in the sixth section. Contrary to OM, the SVR-based metrics are less sensitive to duration and more sensitive to the sequencing, the ordering of the states. The exact behavior of the metric can be adjusted using two parameters. The exponential transformation of time (the b parameter) raises the sensitivity to duration and timing, while subsequence length weighting (a parameter) makes the distance measure more sensitive to the ordering of *complex* sequences.

Our simulations and application have highlighted the difference between OM and SVR metrics. This can be used to justify the use of one or the other

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

metric and to further interpret differences in results produced by different metrics. Finally, it also helps to interpret the structure of the data. If SVR-based distances produce better results, it might be because the data are more structured according to ordering than according to state durations. Therefore, we believe that the SVR family is a useful alternative to alignment-based methods.

Authors' Note

This publication is part of the research works conducted within the Swiss National Centre of Competence in Research LIVES—Overcoming vulnerability: Life course perspectives, which is financed by the Swiss National Science Foundation.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

Notes

1. Part of the content of this article was presented during the Lausanne Conference on Sequence Analysis (LaCOSA'12) at the University of Lausanne on June 6–8, 2012.
2. We will use the acronym OM to refer to the metric and to an algorithm required to calculate the metric.
3. Presuming $\sum$ is countable, the Kleene closure $\sum^*$ must be countable.
4. The elliptical inner product is often used in statistical pattern recognition in the guise of the Mahalanobis distance (see, e.g., Duda, Hart, and Stork 2001; McLachlan 1992). Both the Euclidean distance and the Mahalanobis distance are special cases of the class of Bregman divergences (e.g., Banerjee et al. 2005) that, in general, do not satisfy the triangle inequality.
5. In an unpublished manuscript (Elzinga 2006), it was suggested to use the "Minimum Amount of Shared Time" instead of the inner product of equation (23). As this alternative does not derive from an inner product, we do not deal with this alternative here.
6. TraMineR is freely downloadable from http://mephisto.unige.ch/traminer/.
7. Recently, Bonetti, Piccarreta, and Salford (2013) published a similar approach under the acronym Analysis of Dissimilarity (ANODI).

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

8. In order to reduce computation time, the simulations were computed on unique sequences only by weighting unique sequences by their frequencies (see Studer et al. 2011, for weighting formulae). The sequence aggregation procedure is fully described in Studer (2013) and can also be used for clustering or any other distance-based sequences analysis.
9. Since the OM-metric cannot be varied with a matching coefficient, we varied the substitution costs in the previous calculations by setting these costs equal to $2 - 2m(a, b)$.
10. The data were collected through the Swiss Household Panel (www.swisspanel.ch) in 2002 using a retrospective biographical survey.
11. This operation has no effect on any distance analysis method.
12. The framework presented here allows to use set proximities in NMS.

References

Abbott, Andrew and Alexandra Hrycak. 1990. "Measuring Resemblance in Sequence Data: An Optimal Matching Analysis of Musicians' Careers." *American Journal of Sociology* 96:144-85.

Aisenbrey, Silke and Anette E. Fasang. 2010. "New Life for Old Ideas: The "Second Wave" of Sequence Analysis Bringing the "Course" Back Into the Life Course." *Sociological Methods and Research* 38:430-62.

Apostolico, Alberto and Fabio Cunial. 2009. "The Subsequence Composition of a String." *Theoretical Computer Science* 410:4360-71.

Banerjee, Arindam, Srujana Merugu, Inderjit S. Dhillon, and Joydeep Ghosh. 2005. "Clustering with Bregman Divergences." *Journal of Machine Learning Research* 6:1705-49.

Billari, Francesco C., Dimiter Philipov, and Pau Baizán. 2001. "Leaving Home in Europe: The Experience of Cohorts Born Around 1960." *International Journal of Population Geography* 7:339-56.

Bonetti, Marco, Raffaella Piccarreta, and Gaia Salford. 2013. "Parametric and Nonparametric Analysis of Life Courses: An Application to Family Formation Patterns." *Demography* 50:881-902.

Brzinsky-Fay, Christian, Ulrich Kohler, and Magdalena Luniak. 2006. "Sequence Analysis with Stata." *The Stata Journal* 6:435-60.

Bürgin, Reto and Gilbert Ritschard. 2014. "A Decorated Parallel Coordinate Plot for Categorical Longitudinal Data." *The American Statistician* 68:98-103.

Bürgin, Reto, Gilbert Ritschard, and Emmanual Rousseaux. 2012. "Exploration graphique de données séquentielles." Pp. 39-50 in *Atelier Fouille Visuelle de Données: méthodologie et évaluation*, edited by Hanene Azzag, Bénèdicte Le Grand, Monique Noirhomme, Fabien Picarougne, and François Poulet. Bordeaux, France: Extraction et Gestion de Connaissances (EGC'2012).

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

Chen, Shihyen, Bin Ma, and Kaizhong Zhang. 2009. "On the Similarity Metric and the Distance Metric." *Theoretical Computer Science* 410:2365-76.

Clote, Peter and Rolf Backofen. 2000. *Computational Molecular Biology: An Introduction*. New York: Wiley.

Crochemore, Maxime, Christophe Hancart, and Thierry Lecroq. 2007. *Algorithms on Strings*. Cambridge, UK: Cambridge University Press.

Dijkstra, Wil and Toon Taris. 1995. "Measuring the Agreement between Sequences." *Sociological Methods & Research* 24:214-31.

Duda, Richard O., Peter E. Hart, and David G. Stork. 2001. *Pattern Classification*. 2nd ed. New York: John Wiley & Sons.

Elzinga, Cees H. 2003. "Sequence Similarity - A Non-aligning Technique." *Sociological Methods & Research* 31:3-29.

Elzinga, Cees H. 2005. "Combinatorial Representation of Token Sequences." *Journal of Classification* 22:87-118.

Elzinga, Cees H. 2006. "Sequence Analysis: Metric Representations of Categorical Time Series." Retrieved June 10, 2014 (http://home.fsw.vu.nl/ch.elzinga/Metrics Revision.pdf).

Elzinga, Cees H. 2009. "CHESA 3.1 User Manual." Technical Report, Department of Social Science Research Methods, Faculty of Social Sciences, VU University Amsterdam, the Netherlands.

Elzinga, Cees H. 2014. "Distance, Similarity and Sequence Comparison." Pp. 51/4-73/4 in *Advances in Sequence Comparison: Methods, Theories and Applications*, edited by Philippe Blanchard, Felix Bühlmann, and Jacques-Antoine Gauthier, Life Course Research and Social Policies. Springer, New York.

Elzinga, Cees H. and Aart C. Liefbroer. 2007. "De-standardization and Differentiation of Family Life Trajectories of Young Adults: A Cross-national Comparison Using Sequence Analysis." *European Journal of Population* 23: 225-50.

Elzinga, Cees H., Sven Rahmann, and Hui Wang. 2008. "Algorithms for Subsequence Combinatorics." *Theoretical Computer Science* 409:394-404.

Elzinga, Cees H. and Hui Wang. 2013. "Versatile String Kernels." *Theoretical Computer Science* 495:50-65.

Elzinga, Cees H., Hui Wang, Zhiwei Lin, and Yash Kumar. 2011. "Concordance and Consensus." *Infomation Sciences* 181:2529-49.

Emms, Martin and Hector Franco-Penya. 2012. "On Order Equivalences between Distance and Similarity Measures on Sequences and Trees." Pp. 15-24 in *Mathematical Methodologies in Pattern Recognition and Machine Learning, Springer Proceedings in Mathematics & Statistics*, vol. 30, edited by P. L. Carmona, J. S. Sanchez, and A. L. Fred. New York: Springer.

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

Gabadinho, Alexis, Gilbert Ritschard, Nicolas S. Müller, and Matthias Studer. 2011. "Analyzing and Visualizing State Sequences in R with TraMineR." *Journal of Statistical Software* 40:1-37.

Gauthier, Jacques-Antoine, Eric D. Widmer, Philipp Bucher, and Cedric Notredame. 2009. "How Much Does It Cost?: Optimization of Costs in Sequence Analysis of Social Science Data." *Sociological Methods & Research* 38:197-231.

Gauthier, Jacques-Antoine, Eric Widmer, Philipp Bucher, and Cedric Notredame. 2010. "Multichannel Sequence Analysis Applied to Social Science Data." *Sociological Methodology* 40:1-38.

Gower, John C. 1971. "A General Coefficient of Similarity and some of its Properties." *Biometrics* 27:857-71.

Gower, John C. and Pierre Legendre. 1986. "Metric and Euclidean Properties of Dissimilarity Coefficients." *Journal of Classification* 3:5-48.

Halpin, Brendan. 2010. "Optimal Matching Analysis and Life-course Data: The Importance of Duration." *Sociological Methods & Research* 38:365-88.

Hollister, Matissa N. 2009. "Is Optimal Matching Sub-optimal?" *Sociological Methods & Research* 38:235-64.

Hubert, Lawrence J. and Joel R. Levin. 1976. "A General Statistical Framework for Assessing Categorical Clustering in Free Recall." *Psychological Bulletin* 83: 1072-80.

Inselberg, Alfred. 2009. *Parallel Coordinates: Visual Multidimensional Geometry and its Aplications*. New York: Springer.

Kaufman, Leonard and Peter J. Rousseeuw. 1990. *Finding Groups in Data. An Introduction to Cluster Analysis*. New York: John Wiley.

Lesnard, Laurent. 2008. "Off-scheduling within Dual-earner Couples: An Unequal and Negative Externality for Family Time." *American Journal of Sociology* 114:447-90.

Lesnard, Laurent. 2010. "Setting Cost in Optimal Matching to Uncover Contemporaneous Socio-temporal Patterns." *Sociological Methods & Research* 38:389-419.

Levine, Joel H. 2000. "But What Have You Done for us Lately?" *Sociological Methods & Research* 29:34-40.

Martin, Peter and Richard D. Wiggins. 2009. "Optimal Matching Analysis." Pp. 385-408. In *Sage Handbook of Methodological Innovations*, edited by Malcolm Williams and W. Paul Vogt. Thousand Oaks, CA: Sage.

McLachlan, Geoffrey J. 1992. *Discriminant Analysis and Statistical Pattern Recognition*. New York: Wiley Interscience.

McVicar, Duncan and Michael Anyadike-Danes. 2002. "Predicting Successful and Unsuccessful Transitions from School to Work by using Sequence Methods." *Journal of the Royal Statistical Society. Series A* 165:317-34.

Meyer, Carl Dean. 2000. *Matrix Analysis and Applied Linear Algebra*. Philadelphia, PA: Society for Industrial and Applied Mathematics (SIAM).

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

Moen, Pirjo. 2000. "Attribute, Event Sequence and Event Type Similarity Notions for Data Mining." PhD thesis, Department of Computer Science, University of Helsinki, Helsinki, Finland.

Mooi-Reci, Irma. 2012. "Retrenchments in Unemployment Insurance Benefits and Wage Inequality: Longitudinal Evidence from the Netherlands, 1985-2000." *European Sociological Review* 28:594-606.

Müller, Nicolas, Alexis Gabadinho, Gilbert Ritschard, and Matthias Studer. 2008. "Extracting Knowledge from Life Courses: Clustering and Visualization." Pp. 176-85 in *Data Warehousing and Knowledge Discovery. Proceedings of the 10th International Conference on Knowledge Discovery, DaWaK 2008, Turin, Italy*, edited by Il-Yeol Song, Johan Eder, and Manh Nguyen, volume 5182 of *Lecture Notes in Computer Science*. New York: Springer.

Pollak, Christophe. 2010. "Analyse des parcours de pauvreté: rapport des enquètes longitudinales." *Informations Sociales* 6:106-12.

Pollock, Gary. 2007. "Holistic Trajectories: A Study of Combined Employment, Housing and Family Careers by Using Multiple-sequence Analysis." *Journal of the Royal Statistical Society A* 170:167-83.

Rousset, Patrick and Jean-François Giret. 2007. "Classifying Qualitative Time Series with SOM: The Typology of Career-paths in France." Pp. 749-56 in *Computational and Ambient Intelligence: Ninth International Work-conference on Artificial Neural Networks, IWANN 2007 (LNCS 4507)*, edited by Francisco Sandoval, Alberto Prieto, Joan Cabestany, and Manuel Graña. New York: Springer.

Sankoff, David and Joseph B. Kruskal, eds. 1983. *Time Warps, String Edits and Macro-molecules. The Theory and Practice of String Comparison*. Reading, MA: Addison-Wesley.

Schölkopf, Bernhard and Alexander J. Smola. 2002. *Learning with Kernels. Support Vector Machines, Regularization Optimization, and Beyond*. Cambridge, MA: MIT Press.

Settersten, Richard A. and Karl-Ulich Mayer. 1997. "The Measurement of Age, Age Structuring, and the Life Course." *Annual Review of Sociology* 23:233-61.

Studer, Matthias. 2012. "Étude des inégalités de genre en début de carriére académique á l'aide de méthodes innovatrices d'analyse de données séquentielles." PhD thesis no 777. Faculté des sciences économiques et sociales de l'Université de Genève. Retrieved June 10, 2014. (http://archive-ouverte.unige.ch/unige:22054).

Studer, Matthias. "WeightedCluster Library Manual. A Practical Guide to Creating Typologies of Trajectories in the Social Sciences with R." Technical Report, LIVES Working Papers, 24, Institute for Demographic and Life Course Studies, University of Geneva, Geneva, Switzerland. doi:http://dx.doi.org/10.12682/lives.2296-1658.2013.24.

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015

Studer, Matthias, Gilbert Ritschard, Alexis Gabadinho, and Nicolas S. Müller. 2011. "Discrepancy Analysis of State Sequences." *Sociological Methods & Research* 40:471-510.

Thomsin, Laurence, Jean-Marie Le Goff, and Claudine Sauvain-Dugerdil. 2004. "Genre et étapes du passage à la vie adulte en Suisse." *Espace populations sociétés* 1:81-96.

Tversky, Amos. 1977. "Features of Similarity." *Psychological Review* 84:327-52.

Wang, Hui. 2006. "Nearest Neighbors by Neighborhood Counting." *IEEE Transactions on Pattern Learning and Machine Intelligence* 28:1-12.

Wu, Lawrence L. 2000. "Some Comments on "Sequence Analysis and Optimal Matching Methods in Sociology: Review and Prospect." *Sociological Methods & Research* 29:41-64.

Author Biographies

Cees H. Elzinga is a professor in pattern recognition and vice dean of the Faculty of Social Sciences of the VU University in Amsterdam. He is especially interested in historical demography, life course research, in social networks, in computer science, and in particular in pattern recognition and machine learning. Recent publications appeared in *Advances in Life Course Research*, *Pattern Recognition Letters*, and *Information Sciences*.

Matthias Studer is a postdoc in the Institute of Demographic and Life Course Studies of the University of Geneva and a member of the Swiss NCCR program "LIVES" overcoming vulnerability: life course perspectives." He worked on sequence analysis and gendered career inequalities and recently published on *Discrepancy Analysis in Sociological Methods & Research*.

Downloaded from smr.sagepub.com at Vrije Universiteit 34820 on February 19, 2015